Journal of Network and Computer Applications 79 (2017) 41–67

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Review

Malicious accounts: Dark of the social networks

Kayode Sakariyah Adewole[a,*], Nor Badrul Anuar[a,*], Amirrudin Kamsin[a], Kasturi Dewi Varathan[a], Syed Abdul Razak[b]

[a] Faculty of Computer Science and Information Technology, University of Malaya, 50603 Kuala Lumpur, Malaysia
[b] Faculty of Arts and Social Sciences, University of Malaya, 50603 Kuala Lumpur, Malaysia

ARTICLE INFO

Keywords:
Online social network
Social spam
Malicious behavior
Fake account
Phishing detection
Sybil

ABSTRACT

Over the last few years, online social networks (OSNs), such as Facebook, Twitter and Tuenti, have experienced exponential growth in both profile registrations and social interactions. These networks allow people to share different information ranging from news, photos, videos, feelings, personal information or research activities. The rapid growth of OSNs has triggered a dramatic rise in malicious activities including spamming, fake accounts creation, phishing, and malware distribution. However, developing an efficient detection system that can identify malicious accounts, as well as their suspicious behaviors on the social networks, has been quite challenging. Researchers have proposed a number of features and methods to detect malicious accounts. This paper presents a comprehensive review of related studies that deal with detection of malicious accounts on social networking sites. The review focuses on four main categories, which include detection of spam accounts, fake accounts, compromised accounts, and phishing. To group the studies, the taxonomy of the different features and methods used in the literature to identify malicious accounts and their behaviors are proposed. The review considered only social networking sites and excluded studies such as email spam detection. The significance of proposed features and methods, as well as their limitations, are analyzed. Key issues and challenges that require substantial research efforts are discussed. In conclusion, the paper identifies the important future research areas with the aim of advancing the development of scalable malicious accounts detection system in OSNs.

1. Introduction

Online social networks (OSNs) have emerged as important platforms for people to communicate across the globe. Since the introduction of the first OSN, SixDegree, in 1997 several social networking platforms, such as Facebook, Twitter, and LinkedIn have gained popularity (Heidemann et al., 2012). A recent report from Statista in April 2016 shows a tremendous growth in social network data. Market leader Facebook was the first social networking site whose registered users have surpassed 1 billion and the number of its monthly active users is currently estimated at 1.59 billion (Statista, 2016). The eighth-ranked photo-sharing app, Instagram, had more than 400 million monthly active users. Meanwhile, Tumblr microblogging had over 555 million active users on its network. Twitter microblogging social network was released in 2006 and has attracted more than 320 million monthly active users (Statista, 2016), posting over 500 million tweets per day (DMR, 2015). The popularity of OSNs attracts a lot of attentions among social network users. For instance, organizations leverage social platforms to promote their products and reach out to customers directly on their networks. Celebrities utilize OSN to communicate with their fans. Academia takes advantage of them to record large citations for their articles, and news media distribute their breaking news on these platforms (Cresci et al., 2015; Igawa et al., 2016). Individual also uses social networks to connect with long-lost friends, create text-based contents, publish contents, browse friends' profiles, post photos, share multimedia files, and engage in other numerous social activities. As a consequence, the rapid growth of social networks has triggered a dramatic increase in malicious activities (Fire et al., 2014).

There are numerous malicious activities on social networks including the use of social engineering attack, malware, and spam distribution. Spammers have used social engineering attack strategy to steal the credentials of legitimate users and eventually compromise their accounts (Egele et al., 2015). Information stolen from legitimate users can be used to create Sybil accounts (referred to as fake accounts in this context) in order to deceive the friends of the real users (Bilge et al.,

* Corresponding author.
E-mail addresses: adewole.ks@siswa.um.edu.my (K.S. Adewole), badrul@um.edu.my (N.B. Anuar), amir@um.edu.my (A. Kamsin), kasturi@um.edu.my (K.D. Varathan), syedabdrazak@um.edu.my (S.A. Razak).

http://dx.doi.org/10.1016/j.jnca.2016.11.030
Received 2 September 2016; Received in revised form 2 November 2016; Accepted 28 November 2016
Available online 29 November 2016
1084-8045/

2009) or to send customized spam messages (Ezpeleta et al., 2015; Fire et al., 2014). Facebook social network estimates that 8.7% of its accounts, which amount to 83.09 million do not belong to real users and an estimates of about 1.5% (14.32 million) are undesirable accounts belonging to users that may intentionally spread malicious contents, such as spam messages and suspicious links (Fire et al., 2014). The use of shorten URL on some social media further gives spammers the upper hand to overpower legitimate users and obfuscate their malicious links (Chen et al., 2014). For example, malicious link was used in 2008 to distribute Koobface malware on Facebook and MySpace (Chen et al., 2014). This same malware targeted Twitter users in 2009 (Ostrow, 2009). Thomas and Nicol (2010) studied various malicious characteristics of Koobface. They found that about 81% of users became the victims of this malware during the period of their investigation. The impact of malicious activities on social networks is large on both social network providers and the users as they damage the social trusts among users and undermine the extent of social media.

With the widespread of malicious activities on social networks, the privacy of the users is becoming a major concern (Ge et al., 2014). To reduce the spread of these unwanted behaviors, OSN providers employ a Turing test method, which utilizes CAPTCHA to distinguish automated from human activities. In some cases, they require phone verification to confirm the authenticity of an account. However, CAPTCHA approach is vulnerable to identity clone attack (ICA) allowing attackers to have access to sensitive profile information of their victims (Bilge et al., 2009). Grier et al. (2010) collected large dataset from Twitter and verified every link posted by the accounts in the dataset using blacklist based approach. In blacklist based method, the URL posted by social network user is verified against the popular blacklist APIs, such as Google Safe Browsing, PhishTank, and URIBL. If a URL is detected as suspicious by the blacklist API, the account posted the URL is classified as malicious. However, the key problem with blacklist based approach is that it takes a longer time before a malicious URL is updated in the blacklists and in most cases about 90% of the visitors accessed the malicious page before it is blacklisted (Grier et al., 2010).

Researchers from both industry and academic community have proposed alternative methods. Facebook proposed immune system (Stein et al., 2011) to identify threats on its network (Fire et al., 2014). Facebook also proposed EdgeRank algorithm that assigns a score to the user using some selected features. The limitation of this algorithm is that spammers could form colluding networks and boost their EdgeRank score (Jiang et al., 2012; Zheng et al., 2015). Despite the growth in spam volumes, Twitter lacks an effective spam detection mechanism relying on some rules of thumb to suspend accounts on its network (Twitter, 2016). In the academic community, Wang et al. (2012a) proposed a crowdsourcing method, which relies on human detection to identify fake accounts on social networks. Although this approach provides suitable performance on small data, the problem begins when the number of accounts is very large. This will require substantial human efforts to perform a near-accurate detection. To provide better detection methods, graph-based analysis (Cao et al., 2012; Viswanath et al., 2011) and machine learning approaches have been studied (Lee and Kim, 2014; Yang et al., 2013). For instance, Cao et al. (2012) studied the social structure of users on the network and proposed a SybilRank algorithm. Yang et al. (2015) leveraged friendship invitation graph to identify fake accounts on Renren network. Yang et al. (2013) combined different features to train machine learning classifiers for spam account detection. However, a study conducted by Viswanath et al. (2011) revealed that there is a limit to using only the structure of the social network to effectively detect malicious accounts while the studies on machine learning have faced serious evasion tactics in the hands of intelligent attackers (Yang et al., 2013).

Although, there are many solutions in the literature aim to identify malicious accounts and their behaviors but only a few studies actually survey these approaches. While most of the reviews focused on anomaly detection using graph-based methods (Akoglu et al., 2015; Savage et al., 2014), none of them discussed the features and the state-of-the-art methods used in the previous studies that addressed detection of spam accounts, fake accounts, compromised accounts, and phishing on social networks. For instance, Savage et al. (2014) presented a survey on anomalies detection in a social network with more emphases on graph-based anomaly detection approaches. Xinfang (2013) presented a brief survey of model and techniques for OSNs. They discussed the characteristics of the social network, user group and influence analysis. Heidemann et al. (2012) discussed the characteristics, functionalities, and the impact of using OSN for business activities. Akoglu et al. (2015) presented a comprehensive review of anomaly detection methods specifically targeting graph-based approaches on different networks. It is important to state that this review focuses on the four identified categories through an in-depth analysis of the different features and methods proposed for detecting malicious accounts and their associated behaviors. While graph-based anomaly detection method can be used to detect each of these categories, this paper argues that it is important to pay attention to the several features and methods, which have been specifically employed in these studies. In addition, graph-based anomaly detection is limited to the problem domain under consideration. Therefore, a closely related review to this work is the study presented by (Viswanath et al., 2011), however, their review only addressed studies on Sybil (i.e fake accounts) detection and very few studies were considered in this article. This review is comprehensive covering 65 articles from the literature. The goal is to highlight issues and challenges with existing features and methods utilized for detecting malicious accounts and their behaviors. This paper further highlights the key areas for future research in this domain.

The remainder of this paper is organized as follows: Section 2 presents online social network definitions, categorizations, malicious accounts, and datasets. Section 3 highlights the review methodology. Section 4 discusses the proposed taxonomy based on the different categories of features used for malicious accounts detection. Section 5 presents the taxonomy based on methods, such as crowdsourcing, graph-based and machine learning. Section 6 focuses on the identified issues and challenges. Section 7 presents the proposed framework for malicious account detection. Finally, Section 8 draws the main conclusion. Table 1 describes the various abbreviations used in this paper with their expansions.

2. Online Social Network (OSN)

Social networks emerged from different interdisciplinary fields of studies. The term social network is also used in the fields of social psychology, sociology, statistics, and graph theory to represent a social structure that consists of a set of individuals or organizations with various interactions or relationships among them. With the emergence of the World Wide Web (WWW) and the advancement of technologies, social networks attained a new dimension (Ahmad and Sarkar, 2016; Heidemann et al., 2012). The introduction of SixDegree social networking site in 1997 gave rise to different social networks, such as LiveJournal, MySpace, Facebook and LinkedIn as shown in Fig. 1. This section presents the definitions and categorizations of OSNs, the different categories of malicious accounts, and concludes the section with a detailed discussion of online social network datasets.

2.1. Definitions and categorizations

Following the launch of SixDegree.com, several notable definitions have been used to describe social network. For instance, Boyd and Ellison (2007) defined online social networking site as "web-based service which allows individuals to construct a public or semi-public profile within a bounded system, articulate a list of other users with

Table 1
List of abbreviations with their expansions.

Abbreviations	Expansions
ANN	Artificial Neural Network
API	Application Programming Interface
AUC	Area Under Curve
CAPTCHA	Completely Automated Public Turing test to tell Computer and Human Apart
DM	Direct Message
DOM	Document Object Model
FNR	False Negative Rate
FPR	False Positive Rate
FW	Filter Wall
GMM	Gaussian Mixture Model
GD	Gradient Descent
HC	Hierarchical Clustering
HDFS	Hadoop Distributed File System
ID3	Iterative Dichotomiser 3
MCL	Markov Clustering algorithm
ML	Machine Learning
MLP	Multilayer Perceptron
OSNs	Online Social Networks
PCA	Principal Component Analysis
QAE	Quantum Evolutionary Algorithm
RBF	Radial Basis Function
REST	Representational State Transfer
SMO	Sequential Minimal Optimization
SVM	Support Vector Machine
TNR	True Negative Rate
TPR	True Positive Rate
TSVM	Transductive Support Vector Machine
URL	Uniform Resource Locator
VSNs	Video Social Networks
WWW	World Wide Web

whom they share a connection and traverse their list of connections and those made by others within the system". Adamic and Adar (2005) stated that "social networking service gather information on users' social contacts, construct a large interconnected social network, and reveal to users how they are connected to others in the network". Schneider et al. (2009) defined OSN as a form of online communities among people with common interests, activities, backgrounds, and/or friendships.

Social networks can be categorized according to the purpose they are developed. For instance, Table 2 presents social networks based on their primary objectives. Each of these networks particularly targets a diverse group of users with a specific focus on rendering unique services to their registered users. For example, a social network like Facebook was developed with the prime purpose of providing a private network where users can share their experiences. A network, such as LinkedIn was launched for business purpose. If a researcher wants to make his articles and research activities available to the research community, he may choose to use ResearchGate. This shows that each OSN has unique purpose and functionalities for which the platform is developed to serve the registered users.

2.2. Malicious accounts on social networks

As shown in Fig. 2, accounts used for malicious activities on the social networks can be categorized into two: Fraudulent/career-spamming and compromised accounts. An adversary creates fraudulent accounts for distributing malicious contents, such as embedding malicious links to phishing web pages in order to obtain sensitive information from the victim. By collecting a large number of legitimate users information as well as information about friends of friends on the network, an adversary can create Sybil or fake accounts by forging the existing users' identities. The fake identity is used to overpower legitimate users and undermine the trust relationship on the social network in order to perform different malicious activities. These activities may include social spamming, drive-by-download, and private data harvesting (Chen et al., 2014). To ensure that these fraudulent accounts stay for a longer period on the network, attackers sometimes equipped them with automated characteristics making them have the ability to post contents that mimic real users. Fake account on the social network has turned into a multimillion dollar business, where several fake accounts are advertised on the Internet for those who wish to boost their account reputation. A recent report indicates that accounts of celebrities, politicians, and popular organizations featured a suspicious increase in fake accounts (Cresci et al., 2015). This kind of underground activities further damage the social reputation and indeed expose the social network to a lot of risks. For instance, platforms such as Intertwitter (http://intertwitter.com/) offered 10,000 fake followers accounts at the rate of $79, giving spammers the opportunity to embed themselves easily within the network of legitimate users. Fake accounts are now offered in large volumes, varying from thousands to millions (Zhang and Lu, 2016). These bogus accounts and their malicious links infringe on the normal social network trust and disrupt the media for effective social interaction.

On the other hand, compromised account is an account hijacked from legitimate users using a strategy such as social engineering attack to deceive legitimate users by clicking on links to phishing web pages. Previous study shows that compromised accounts are more useful to spammers than career-spamming accounts since they enable spammers to leverage the existing trust relationship between the accounts and the social network providers (Egele et al., 2015). Once hijacked, compromised accounts will start experiencing sudden changes in posting patterns since it will be difficult for spammers to completely maintain the normal posting behaviors of the real owners (Ruan et al., 2016). An example of the sudden posting patterns may include the use of compromised accounts to spread spam messages, such as favorite, direct message (DM) that contains malicious links, and click spam. The victim may also be engaged with different malicious campaigns, such as pornography, fake news, and donation scam. Social network providers, as well as users, are willing to detect and completely remove these malicious accounts; however, the fight between attackers and the detection system has been a game of cat and mouse. Once an approach is used to identify misbehaving accounts, attackers devised a new strategy to possibly evade the detection approach. This behavior posed a lot of challenges to the existing detection systems.

2.3. Impact of malicious activities in OSNs

Due to the increase in the number of malicious accounts created on the social networks on a daily basis, the impact of malicious activities has increased drastically. According to the Nexgate report in 2013, malicious activities on social networks, such as spam distribution has risen to about 355% in the first half of the year 2013. This report further concludes as follows: (1) 5% of all apps on social media are for spam purpose. (2) Malicious accounts on social networks post a large volume of contents and more quickly than the normal accounts. (3) An adversary often distributes spam contents to at least 23 social network accounts. (4) For every seven (7) new social media accounts created there are at least five spammers. (5) 15% of all social spam messages contains a URL to spam content, malware, or pornographic websites (Nguyen, 2013). The findings revealed that malicious accounts have been used to steal the identities of a large number of legitimate users across different social networks. There has been a substantial increase in the number of identity fraud cases in the past few years. In a recent report by Javelin Strategy and Research, the total number of identity fraud victims has grown to about 13 million per year and around $112 billion has been stolen in the past six years (Javelin Strategy and Research, 2016). Social spammers make about $200 million every year constituting to a loss in social trust, productivity and profit. In fact, the growth in pornographic spam has doubled on most popular social

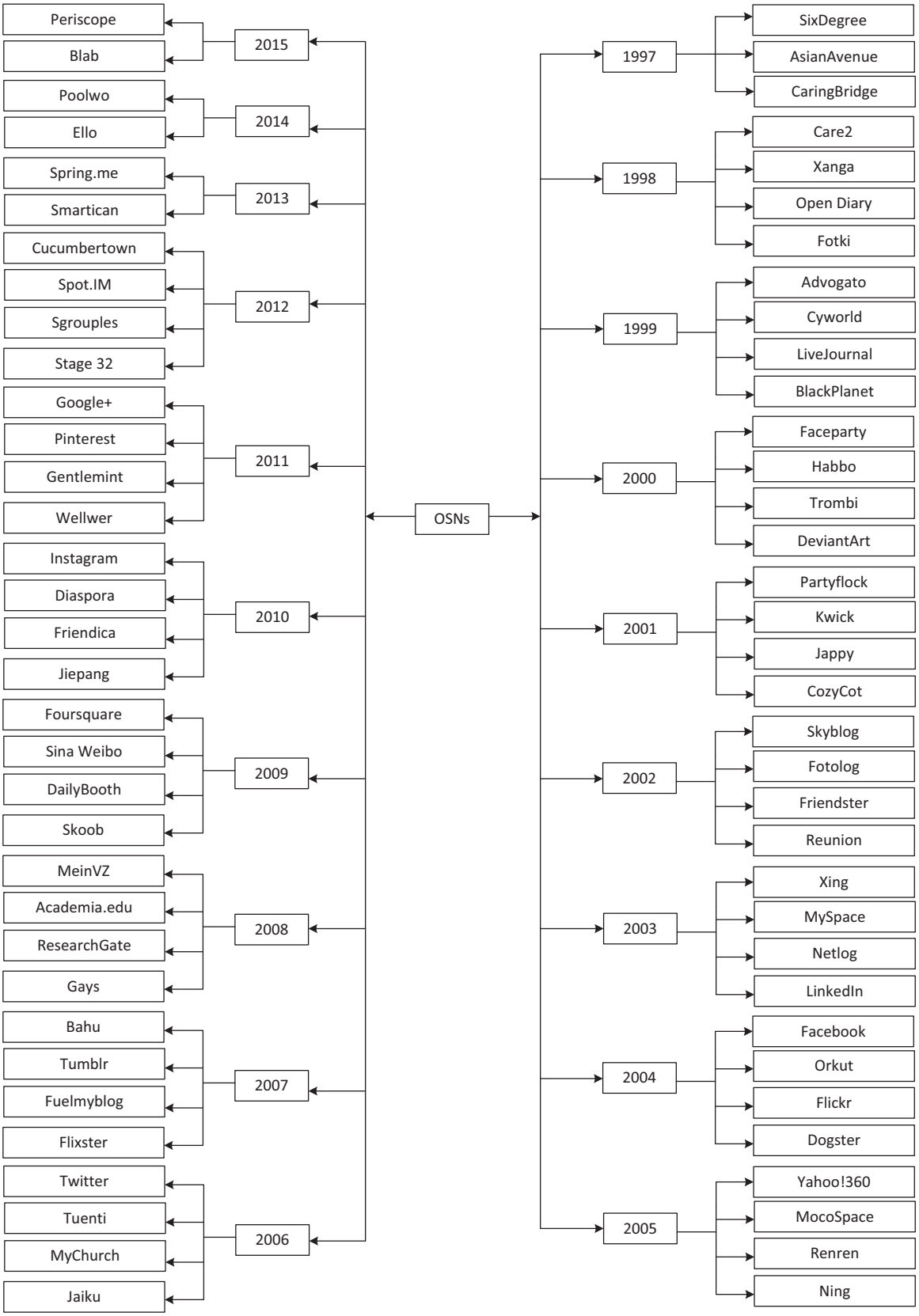

Fig. 1. Some examples of OSNs since 1997.

Table 2
Types of social networks based on purpose.

Type	Description	Example
Private social networks	These social networks are specifically developed for private use.	Facebook, MySpace
Business social networks	Introduced for business purpose.	Xing, LinkedIn
Academic social networks	These are developed for academic researchers.	Academia.edu, ResearchGate
Microblogging and News update	They provide a platform for sharing latest updates about what people are doing.	Twitter, Tumblr.
Video sharing social networks	These are developed for sharing different kinds of videos including tutorials and news.	YouTube, Flickr
Instant messaging social networks	These are cross-platform messaging applications developed for sharing video, text, images and audio contents.	Skype, WhatsApp
Event social network	The OSNs connect customers with events, entertainments, and movies.	Eventful, Zvents
Location-based social networks	These OSNs help people to look around for perfect places to go with their friends.	Foursquare, Apontador

networks aiming to entice users to click malicious links or download malware (Ab Razak et al., 2016). Since the impact of malicious activities is rising, it is important to remove accounts that pose a threat to legitimate users on the network. The subsequent sections discuss the several efforts that have been made in the literature to achieve this objective.

2.4. Online social network dataset

This section categorizes the datasets used in the previous studies that deal with detection of malicious accounts into two main groups: graph and non-graph. The first category modeled social network as a graph represented by nodes and edges. The second category contains different features extracted from social network data, which are used to build a detection system. It is important to state that there are several publicly available graph datasets for social network research as shown in Table 3. The most prominent are those compiled by Stanford University social network research community (Leskovec, 2015). The datasets contain many social network graphs including Facebook, Twitter, and LiveJournal. However, some researchers also developed web crawlers to collect private graph data from a social network of interest.

In some cases, researchers evaluated their proposed models using synthetic social graph generated by applying Barabasi-Albert preferential attachment model. This model assumes that social network is scale-free and it follows a power law distribution. For instance, SybilRank (Cao et al., 2012) and community detection algorithms (Viswanath et al., 2011) were evaluated using synthetic datasets in addition to publicly available real-world graph datasets.

Due to the fear of violating users' privacy, some of the public non-graph datasets have been secured with passwords. In addition, they contain limited numbers of users' attributes released for research purpose. This constraint forces researcher to develop web crawlers to collect private data using the API provided by the social network providers (Alsaleh et al., 2014; Yang et al., 2013). For instance, Twitter provides REST API and Streaming API to collect tweets, network data, and other information from its platform. Facebook also provides Facebook Graph API to get data in and out of the Facebook social network.

For example, Yang et al. (2013) collected around 500,000 Twitter accounts with over 14 million tweets posted by them using Twitter API. They applied blacklist and honeypot techniques to identify 2000 Twitter spammers. Conti et al. (2012) developed a sensing application embedded within the Facebook account to collect users statistical information. Chu et al. (2012a) applied Twitter API to collect more than 500,000 users data with over 40 million tweets posted by them. The majority of the studies reviewed in this paper evaluated their proposed models using private datasets. For this reason, it is very difficult to compare existing studies based on the performance metrics employed, since this comparison will introduce bias in the review. Therefore, this paper focuses on categorizing existing studies based on the features and methods used. The subsequent section discusses the review methodology, which describes how the related articles considered in this review were gathered.

3. Review methodology

Since a wide range of malicious activities exists on the social network, this paper focuses on four main study areas: spam account, fake account, compromised account, and phishing detection. For the purpose of clarity, it is important to state the characteristics of the different malicious categories considered in this review as well as how the related studies were grouped under each category. The process starts with the construction of search criteria to retrieve relevant articles from four digital libraries: ACM, IEEE, ScienceDirect, and Springer (see Table 4). The search was performed using the following defined search terms: "fake account" OR "fake profile" OR "Sybil" OR "spam account" OR "social spammer" OR "spam" OR "compromised" OR "phishing" OR "malicious URL" AND "online social networks". The review considers the year of publication of the articles retrieved from 2006 to 2016 and examines their title, abstracts, keywords, and conclusion. Articles that are outside the scope of online social net-

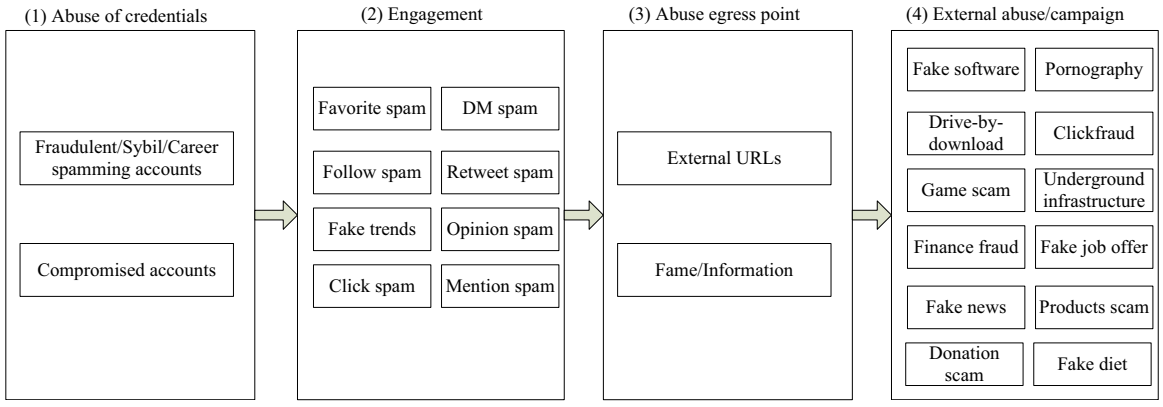

Fig. 2. Abuse of social network accounts for spamming, phishing, and their campaigns.

Table 3
Public datasets.

Type	Category	Name	Description	Web address
Public	Graph	Stanford large network dataset	The dataset is organized into different social graphs, such as ego-Facebook, ego-Twitter, wiki-Vote, com-DBLP, com-Youtube etc.	https://snap.stanford.edu/data/
Public	Non-graph	Deceptive Opinion Spam Corpus	Contains 400 truthful positive reviews from TripAdvisor, 400 deceptive positive reviews from Mechanical Turk, 400 truthful negative reviews from Expedia, Hotels.com, Orbitz, Priceline, TripAdvisor and Yelp, and 400 deceptive negative reviews from Mechanical Turk. These datasets consist of 20 reviews for the most popular Chicago hotels.	http://myleott.com/op_spam/
Public	Non-graph	BibSonomy	This dataset is part of ECML/PKDD Discovery Challenge 2008. The dataset is organized into seven files: tas, tas spam, bookmark, bookmark spam, bibtex, bibtex spam, and user.	www.kde.cs.uni-kassel.de/ws/rsdc08
Public	Non-graph	Tweets2011 Corpus	This dataset is part of the TREC 2011 microblog track. It contains 16 million tweets sampled between Jan. 23rd to Feb. 8th, 2011. It is available for individual download at NIST website. However, this dataset cannot be distributed to other researchers due to privacy issue.	http://trec.nist.gov/data/tweets
Public	Graph and Non-graph	FakeProject	The dataset was released by MIB project hosted at Institute of Informatics and Telematics (IIT) of the Italian National Research Council (CNR). The dataset is organized into five groups: TFP, E13, INT, FSF, and TWT. The dataset requires password due to users' privacy.	http://mib.projects.iit.cnr.it/dataset.html

Table 4
Total number of articles.

Database source	Number of articles
ACM	18
IEEE	18
Science direct	15
Springer	14
Total Articles	**65**

works, such as studies on email spam detection, as well as articles that did not focus on developing a detection model to identify malicious accounts were removed. All articles on spam account and spam message detection are grouped under spam account category. Articles on fake account and Sybil detection are grouped under fake account category. Table 5 provides the details of each of the four categories with the total number of articles considered. In total, 65 articles were selected for the review covering a period of eleven (11) years from 2006 to 2016 as shown in Fig. 3. Fig. 4 shows the distribution of these articles based on the methods used: crowdsourcing, graph-based, and machine learning (ML). This figure shows that majority of the previous studies on malicious accounts detection utilized machine learning method. Throughout this paper, the behavior reveal by these categories is referred to as malicious behavior in social networks. The majority of the previous studies on malicious account detection in online social networks focused on spam and fake account detection with little studies on compromised and phishing detection. In fact, the authors of the recent study on compromised account detection (Ruan et al., 2016) emphasized that their paper is the second article on compromised account detection specifically for online social networks.

4. Taxonomy based on feature analysis

Generally, malicious accounts detection on social networks can be conceptualized using the generic framework in Fig. 5. The inputs to the detector originate from the previously discussed dataset, which may be adjacency matrix or set of features. The input may require data preprocessing, such as removal of accounts with a small number of connections or stringent preprocessing like extraction of N-gram features from the messages posted by the accounts under consideration. The preprocessed data is passed to malicious account detector, which produces output in form of class and rank. The class may be view as spammer or legitimate, compromised or normal, Sybil or non-Sybil nodes, malicious or legitimate and so on. The rank indicates the probability that a given account belong to the final class label. After producing the final class label, the detector is evaluated using the popular evaluation metrics, such as precision, recall, F-measure, accuracy, and Area under Curve (AUC). During the detection, different features and methods have been considered with the goal of identifying the class of a given account or the class of the message sent by this account owner.

Features used in the previous studies for detecting malicious accounts in social networks can be broadly merged under three main analyses: social network analysis, content/behavioral analysis, and hybrid analysis as shown in Fig. 6. This section discusses each of these categories and highlights the different features proposed in the literature to identify malicious accounts and their behaviors in OSNs.

4.1. Social network analysis

Social network analysis involves analyzing the topological social structure of the accounts within the network or extracting discriminative network features to detect malicious users. Some studies focused on analyzing the structure of the users on the network (Cao et al., 2012; Mulamba et al., 2016) while others concentrated on extracting network

Table 5
Malicious categories consider in this paper.

Category	Description	Total articles
Spam account	Spam account distributes unsolicited messages on the social network. This account may represent career-spam account created by spammers. Since spam account has a high probability of posting spam messages, the studies on spam account detection and spam message detection were grouped together.	33
Fake account	Sybil or fake account is a scam account created by malicious users for different malicious activities, such as spamming, cyberbullying, and malware distribution. An attacker may create this fake account by copying the basic profile information of his victim. In some cases, a pseudo-name can be used to create such account with the help of an automated program. Fake account can also be referred to as Sybil or cloned account as used in some studies.	23
Compromised account	Compromised accounts are legitimate accounts hijacked from the real owners. Once the account is hijacked, it can be used to engage the victim with different vulnerabilities, including spamming, cyberbullying, and spreading of malware.	2
Phishing	Phishing is not an account but a strategy used by cybercriminals to post malicious URLs in order to lure their victims to malicious web pages. This attack collects sensitive information from the victims, such as login credentials and sensitive profile data. The studies that focus on phishing or malicious URL detection are grouped under this category.	7

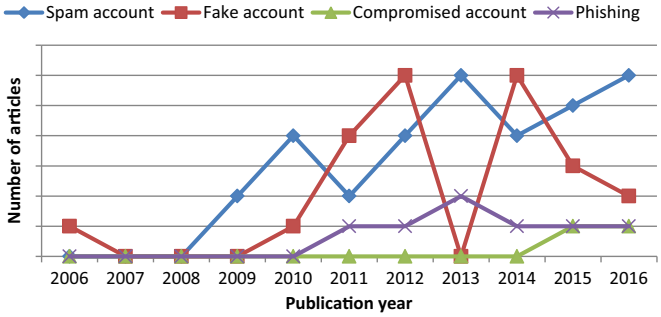

Fig. 3. Annual distribution of journal articles for the different malicious categories.

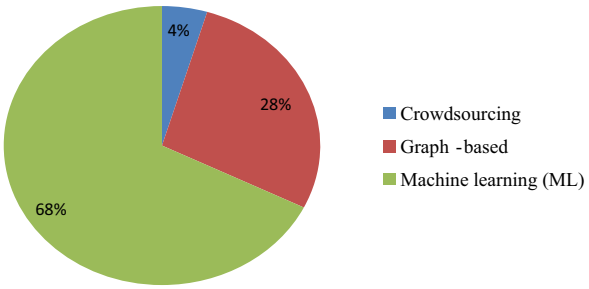

Fig. 4. Distribution of journal articles based on methods.

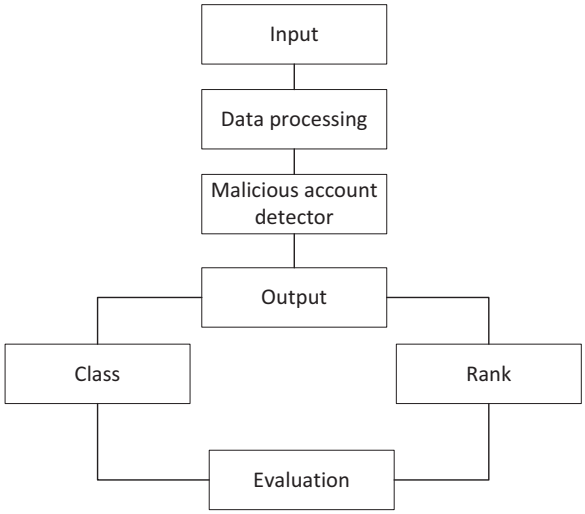

Fig. 5. Generic framework for malicious account detection.

features rather than studying the topological structure in details. Such network features include community-based features (Bhat and Abulaish, 2013; Bhat et al., 2014) or features based on neighborhood and centrality (Almaatouq et al., 2016; Yang et al., 2013).

4.1.1. *Social network structure*

An example of social structural analysis is presented by (Stein et al., 2011) in Facebook Immune System, which makes the assumption that although malicious users may create a large number of fake identities on social networks, however, it will be difficult to establish arbitrarily large number of social relationships to legitimate accounts. This makes them poorly connected to the network when compare with legitimate accounts. This assumption was adopted to develop many Sybil defense algorithms, such as SybilGuard (Yu et al., 2006), SybilLimit (Yu et al., 2008), SybilRank (Cao et al., 2012), and GateKeeper (Tran et al., 2011) believing that fake account will require significant trustworthy social ties to appear legitimate within the network. This feature is analyzed to identify densely connected Sybil regions (Viswanath et al., 2011). For instance, Tran et al. (2011) leveraged random expander graph property and improved ticket distribution technique to study the structural connection of social network users. The proposed GateKeeper algorithm applied different randomly chosen points to run the ticket distribution and merges the outcomes to perform a decentralized node admission control. During node admission, a node that acts as a ticket source distributes a certain number of tickets on the network until a considerable proportion of the honest nodes receives some tickets. The ticket distribution follows a breast first search approach where each node is placed at a breast first search level according to its shortest path distance from the ticket source. The ticket source splits the tickets and distributes them to its neighbors. A single ticket is kept by each node on the network and the remaining tickets are propagated to the nodes in its neighbors at the next level. If a node does not have outgoing connections to the next level, the node simply destroy the remaining tickets. This process continues until no ticket remains to distribute.

However, the effectiveness of this detection approach is limited by this assumption and several experiments have proven the weakness of this approach (Elyashar et al., 2013; Viswanath et al., 2011; Yang et al., 2015). For instance, a social network like Twitter allows a unidirectional user binding, which permits an account to follow anyone on the network without the followee prior consent. Although the followee may decide to block the follower, however, in reality, the majority of them follow back for the sake of courtesy. This behavior allows malicious accounts to add more legitimate users on his network (Hu et al., 2013). In addition, fake accounts are now sold in thousands on the Internet allowing Sybil to embed seamlessly within the network and appear as legitimate accounts. Viswanath et al. (2011) analyzed various Sybil detection algorithms by decomposing Sybil defense approaches. The study revealed that Sybil defense algorithms operate by implicitly ranking nodes (i.e accounts) based on how well they connect to a trusted node. Accounts with a strong connection to the legitimate users

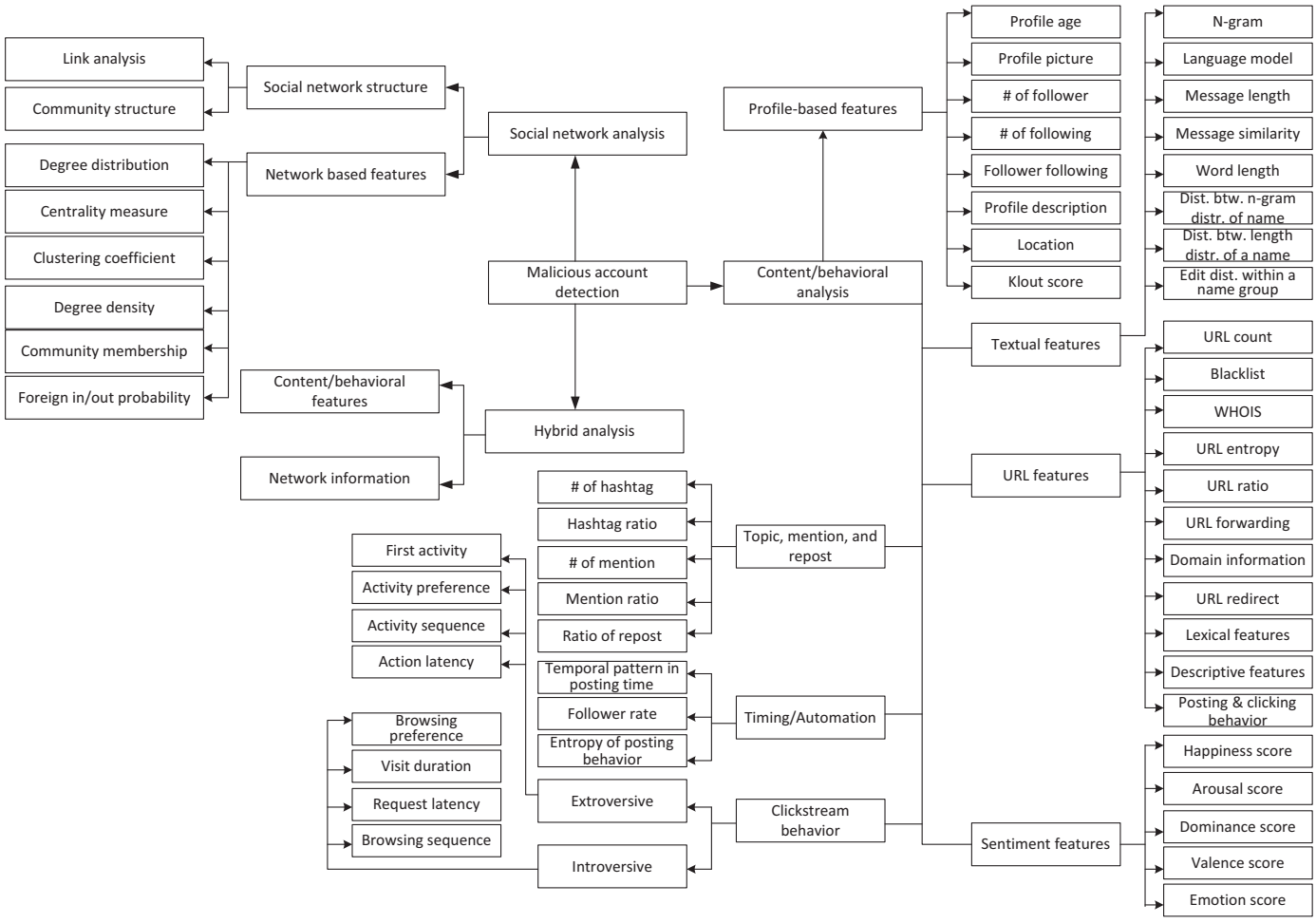

Fig. 6. Taxonomy based on feature analysis.

are given a higher rank score and are deem to be more trustworthy. Section 5 provides a detailed discussion of the ranking method under Trust propagation.

Since the assumption used by the early social network structural Sybil detection algorithms is very weak in providing efficient fake account detection system, algorithms like SybilRadar (Mulamba et al., 2016) and VoteTrust (Yang et al., 2015) deviated from this assumption. For instance, SybilRadar leveraged the assumption that attackers can create as many fake accounts as possible and they can establish significant large links to legitimate accounts. VoteTrust exploited friendship invitation relationship between fake accounts and legitimate users using request invitation graph to identify malicious users. However, a small temporal change in Sybil behavior will break the detection approaches in these studies. This accounts for why some researchers explored the possibility of extracting network features and content/behavioral based features to train machine learning classifiers. The studies in this area assumed that it would be difficult for an attacker to break all the identified features. The authors of SybilRadar algorithm also admitted that the use of some account attributes might improve SybilRadar performance.

In addition to studying the link structure of the users on the network, few studies also relied on analyzing the community structure of the social network to rank accounts and identify the community of malicious users (Liu et al., 2015; Mulamba et al., 2016; Viswanath et al., 2011). However, an intelligent adversary may forge the connectivity of the controlled fake accounts to imitate the network community structure of the portion of the social network exhibited by normal users. This tactic would make it difficult for methods relying on community analysis to effectively detect malicious accounts.

4.1.2. Network based features

Another line of research using network analysis involves the identification of effective network features to detect malicious accounts. Features such as degree distribution, centrality measures, clustering coefficient, degree density, and community membership has been widely studied (Almaatouq et al., 2016; Bhat and Abulaish, 2013). For instance, Bhat and Abulaish (2013) extracted different community-based features from a user's social connections to train machine learning classifiers to detect social spammers. The study reveals that community features, such as total in/out ratio, core node, community membership, foreign out-degree, foreign in/our ratio, foreign out-link probability, reciprocity, and foreign out-link grouping are effective for social spammer detection. Centrality measures like betweenness and closeness, as well as degree computation such as weighted in and out degree, degree density, weighted bidegree, and density of relative edges of both mention and followers networks have played a key role in spammer detection (Almaatouq et al., 2016; Yang et al., 2013). Fire et al. (2014) studied the connection strength between the pair of accounts on Facebook to detect fake accounts, which may pose a risk to the legitimate user and then restrict these set of accounts from accessing the private information of the legitimate user. They proposed a number of network-based features, such as mutual friend and the number of common group between a pair of accounts. They further defined a global connection strength heuristic function capable of identifying fake accounts on the Facebook network.

Apart from the fact that an adversary can evade some network-based features, such as bidirectional links, and bidirectional link ratio (Mulamba et al., 2016; Yang et al., 2013), other challenges have centered on how to deal with the computational complexity when

extracting network based features for large social networks (Fire et al., 2014). Several studies revealed that some network features are expensive for attackers to evade. For example, features such clustering coefficient, betweenness centrality, and following to media neighbors' followers are more robust for identifying social spammers (Almaatouq et al., 2016; Yang et al., 2013).

4.2. Content/behavioral analysis

Content/behavioral analysis involves identification of effective features outside the social connections of users. The studies in this category assumed that the content generation pattern of malicious accounts is different from legitimate users. Thus, extracting many features around this behavior could help distinguish malicious accounts from legitimate ones. One of the advantages of using content/behavioral analysis is that it can be easily encoded into features. These features are provided as input to machine learning algorithms, which can learn the signature of malicious and legitimate activities. Thus, it allows classification of accounts based on the observed behaviors. In this domain, many classes of features are commonly employed to represent users' behaviors as shown in Fig. 6 including the use of profile-based, textual, URL, timing/automation, topic, mention, and reposting behaviors, sentiment as well as clickstream features like extroversive and introversive behaviors.

4.2.1. Profile-based features

Profile-based features involve studying the basic profile information of an account on the network. Studies that utilize profile-based features established that by analyzing profile information of an account, such as profile age, profile picture, number of follower, number of following, follower-following ratio, profile description, geolocation, Klout score, account verification status, listed count, and total number of tweets posted; it is possible to distinguish malicious accounts from legitimate ones (Alsaleh et al., 2014; Chu et al., 2012a; Main and Shekokhar, 2015; Miller et al., 2014). The study conducted by Benevenuto et al. (2009) shows that the use of user behavioral attributes like the total view of tag videos, total ratings of tag videos, and user rank can identify social spam accounts and content promoters on social networks that support video sharing, such as YouTube. Chu et al. (2012a) studied profile-based features to distinguish human accounts from the accounts controlled by an automated program called socialbot. Studies that utilized profile-based features also combined these features with other categories, such as textual, URL and so on. For instance, Stringhini et al. (2010) created honeypot accounts on three social networks: Facebook, MySpace, and Twitter. They logged every user's activities through the honeypot accounts and extracted profile-based and URL features to identify spammers. The issue with content/behavioral features has centered on how to deal with the evasion tactics of malicious users. For instance, profile age was identified as a discriminative feature based on the findings that account of malicious users usually exhibits short profile age (Almaatouq et al., 2016; Lee and Kim, 2014). However, Egele et al. (2015) established that majority of accounts used for spamming on social networks are compromised accounts which are more valuable to spammers due to the pre-established trust relationship. This is similar to the findings in (Gao et al., 2010; Grier et al., 2010) on Facebook and Twitter networks.

4.2.2. Textual features

The use of textual or content features, such N-gram, language model, message length, message similarity, and word length has been studied (Balakrishnan et al., 2016; Gani et al., 2012; Harsule and Nighot, 2016; Martinez-Romo and Araujo, 2013). N-gram based system called Filter Wall (FW) capable of filtering messages in a user's timeline was developed to build user's N-gram profile (Harsule and Nighot, 2016). From this N-gram profile, a similarity distance metrics is applied to categorize wall posts as spam and non-spam messages. Martinez-Romo and Araujo (2013) introduced features based on language model from the textual content of a user's tweets. The feature studied the divergent of textual information of malicious and normal tweets. A language model is a statistical model for text analysis, which is based on a probability distribution over pieces of text, indicating the likelihood of observing these pieces in a language. Usually, the real model of a language is unknown and is estimated using a sample of text representative of that language. Different texts can be compared by estimating models for each of them, and analyzing the models using well-known methods for comparing probability distributions (Martinez-Romo and Araujo, 2013). The authors examined the use of language in different entities, such as a topic, a tweet, and the external page linked from the tweet. It was established that the language model of a legitimate tweet is more likely to be different from the spam tweet. They applied Kullback–Leibler divergence, which is an asymmetric divergence measure adopted from information theory, between respective language models of the entities considered to measure how bad the probability distribution of one language model deviate from other. Based on this assumption, the authors exploit the divergence between the language models to classify tweet as spam or non-spam. This approach has been shown to work well for detecting malicious tweets in trending topics. However, it requires the knowledge of some external contents that may introduce other computational costs. Gani et al. (2012) extracted several features from user's messages including average words length, average message length, average number of words per message, the ratio of uppercase letters, the ratio of short words per message, average number of short words, standard deviation and variance of special characters to detect social spam.

Studies that examined the content/behavioral characteristics of malicious accounts applied off-the-shelf machine learning algorithms to check the effectiveness of the extracted features in distinguishing malicious from legitimate accounts (Lee and Kim, 2014; Martinez-Romo and Araujo, 2013). For instance, Lee and Kim (2014) applied different name-based features, such as distance between unigram/bigram distribution of a name group, edit distance within a name group, distance between length distribution for a name group, and distance between position-wise unigram distribution to train a Support vector machine (SVM) classification algorithm. One of the identifiable key issues with textual features lies in the computational complexity. For example, feature such as N-gram analysis may require several preprocessing steps, which can introduce more computational costs. Section 6 discusses the evasion tactics of textual or content features.

4.2.3. URL features

A large body of studies examined URL features to analyze the URL posting patterns between malicious and legitimate users. A URL is a link embedded within a user's post with an attempt to redirect users to an external page. Malicious users can use this strategy to distribute malicious links and engage victim with fake advertisements. For instance, some studies found that forwarding patterns of URLs, domain, and lexical features are effective for detecting malicious URLs in a user's post providing the opportunity to mine URL posting patterns of malicious and legitimate accounts (Cao et al., 2016; Chen et al., 2014; Lin et al., 2013). While some socialbots are created to post malicious contents on social networks, others mimic the posting patterns of legitimate users. This strategy has been witnessed in a democratic setting where malicious socialbots artificially inflate support for political candidate and abuse the outcome of elections (Ratkiewicz et al., 2011).

The use of URL features, such as dash count in the hostname, longest domain name, domain rank, URL domain age, and URL count was explored in (Chen et al., 2014) to identify malicious links. By analyzing the results of four experiments using a different combination of URL domain anomaly features, Chen et al. (2014) obtained a good classification performance. Aggarwal et al. (2012) built ground truth dataset by analyzing the outcome of different blacklist APIs. They

combined many content/behavioral features including URL features based on WHOIS and URL redirection status. However, the use of WHOIS and URL redirection require the need to query some contents or Internet host-based information, which limit their application in real-time detection of millions of URLs encountered on the social network on a daily basis (Cao and Caverlee, 2015; Lin et al., 2013). To address the problem of querying host-based information, lexical and descriptive URL features were adopted (Lin et al., 2013). The first feature describes the lexical information of the links while the second feature represents some statistical attributes. The lexical features based on the words that appear in the URLs capture the dynamic nature of the links. The static nature of the links is captured by the descriptive features, which rely on the assumption that the characteristics between legitimate and malicious links rarely varied. For instance, phishing websites sometimes utilized related symbols or letters, such as representing the lower case of letter 'L′ with the digit '1′ in order to mislead the target legitimate users. Thus, the websites may have certain statistical information, such as the consecutive relationship of digits and alphabets. Using this assumption, some lexical and descriptive features may be extracted from URLs and use to train classification algorithms (Lin et al., 2013). The study further shows that lexical features are more efficient than descriptive features, but they can only work within a short period. In addition, the descriptive features are less effective but they can be used for a longer period. However, malicious users may continue to change their posting behaviors and try to act like normal users, which can lead to an increase in false positive (Wu et al., 2016). Social media providers will rather prefer not to detect malicious accounts than to continue bothering legitimate users with false accusations.

4.2.4. Topic, mention, and retweet

The majority of social networks allow users to use varieties of tools for communication, such as grouping of messages using topic, sending messages directly to a specific target user, and reposting a user's message. For instance, in Twitter social network, a user can use "#" symbol to indicate the topic of a post. A topic that receives many attentions will eventually become a trending topic. #Women's World Cup and #Justin Bieber are examples of the two popular trending topics on Twitter in 2011 (Chu et al., 2012a). In a similar manner, "@" symbol can be used to forward message directly to a target user on Twitter (e.g @obama). The retweet function allows users to repost messages that appear on their timeline or through specific search keywords. It is believed that malicious users can hide behind trending topics or bypass any requirement for social connection with legitimate accounts by simply use mention function to reach their target victims (Almaatouq et al., 2016). Studies have established that some malicious accounts are equipped with automation capability to repost messages from legitimate users in order to make their account appear legitimate (Cao et al., 2016). By studying the posting behavior of malicious users considering the three aforementioned tools, experiments have shown that malicious accounts tend to post messages with many hashtags and mentions (Alsaleh et al., 2014; Gupta and Kaushal, 2015). Therefore, features such as the number of hashtag, the number of mention, the number of post retweeted as well as their ratios were considered in (Chu et al., 2012a, 2012b). However, the continuous changes in malicious account behavior have posed many challenges to the system relying on this approach.

4.2.5. Timing/automation

Due to the automation capability of some malicious accounts, researchers have studied the temporal posting patterns of malicious and legitimate accounts. For instance, the use of entropy component, which employs tweeting interval as a measure of behavior complexity to detect periodic and regular timing as an indicator of automation has been studied (Chu et al., 2012a). Features like following rate, tweeting rate, API ratio, API URL ratio, and API tweet similarity, which considered the posting time and tweets posting source can be used to identify malicious accounts. Because of the relatively high cost of manually operating a large number of spam accounts, some spammers designed a custom program using API to spread spam messages. Therefore, by studying the source of messages, it is possible to identify malicious accounts. However, Yang et al. (2013) established that features such as API ratio, API URL ratio, and API tweet similarity provide good discriminative performance and can be used to identify spammers on Twitter.

4.2.6. Sentiment features

Sentiment analysis deals with the process of categorizing opinions expressed within a piece of text to determine the attitude or opinion of the writer towards a specific topic or product. It has been shown that malicious accounts used for cyberbullying can concentrate on specific keywords to spread aggressive spam messages (Ferrara et al., 2014; Galán-García et al., 2014). For instance, a message such as "*if you don't follow me you will die, follow me now*" has been used by spammers to spread cyberbullying contents as a strategy to lure the target victims to accept their friendship requests (Galán-García et al., 2014). Extracting sentiment features, such as happiness, arousal, dominance, valence, and emotion scores can help in identifying spammers accounts used for cyberbullying in social networks (Ferrara et al., 2014).

4.2.7. Clickstream behavior

One of the import accounts to an adversary is the account hijacked from legitimate users. Such compromised account will be difficult to distinguish from career-spamming accounts due to its initial legitimate characteristics. To detect compromised accounts, researchers have presented behavioral based features, which analyzed the clickstream characteristics of social network accounts (Ruan et al., 2016). A model of the normal user is developed by considering some of the user's posting patterns over a specific period. This model is then compared with subsequent user behavior to ascertain if the account has been compromised. To effectively build this behavioral profile, clickstream features such as extroversive and introversive behaviors have been studied (Ruan et al., 2016). Extroversive behaviors consider characteristics, such as the first activity the account engages in. While many users may start their social activity by randomly accessing their friends' timelines, others start by liking the posts that appear on their own timeline. Extroversive behavior also includes activity preference, activity sequence, and action latency. Conversely, introversive clickstream behaviors include browsing preference, visit duration, request latency, and browsing sequence. Since clickstream features require extensive study of the user's behavioral patterns, it is difficult to efficiently capture all the normal user's clickstream behaviors. A slight deviation in a normal user's behavior can increase the false positive of the detection system because such behavior will be classified as malicious.

4.3. Hybrid analysis

The presence of many platforms for underground markets where it is possible for malicious users to purchase a large number of followers to boost their fake accounts has hindered the effectiveness of relying on content/behavioral analysis. For example, underground markets, such as *BuyTwitterFriends.com* or *TweetSourcer.com* provides cheap services to purchase fake followers allowing malicious account to appear legitimate (Yang et al., 2013). Since malicious users on the social network have devised strategies to make their accounts appear normal, some studies exploit the assumption that combining both content/behavioral and network information could help identify these misbehaving users. For example, Yang et al. (2013) analyzed the effectiveness of combining network and content information to detect spammers on Twitter. They found that content-based analysis using profile-based, textual, reposting, URL, topic and mention features are very weak in

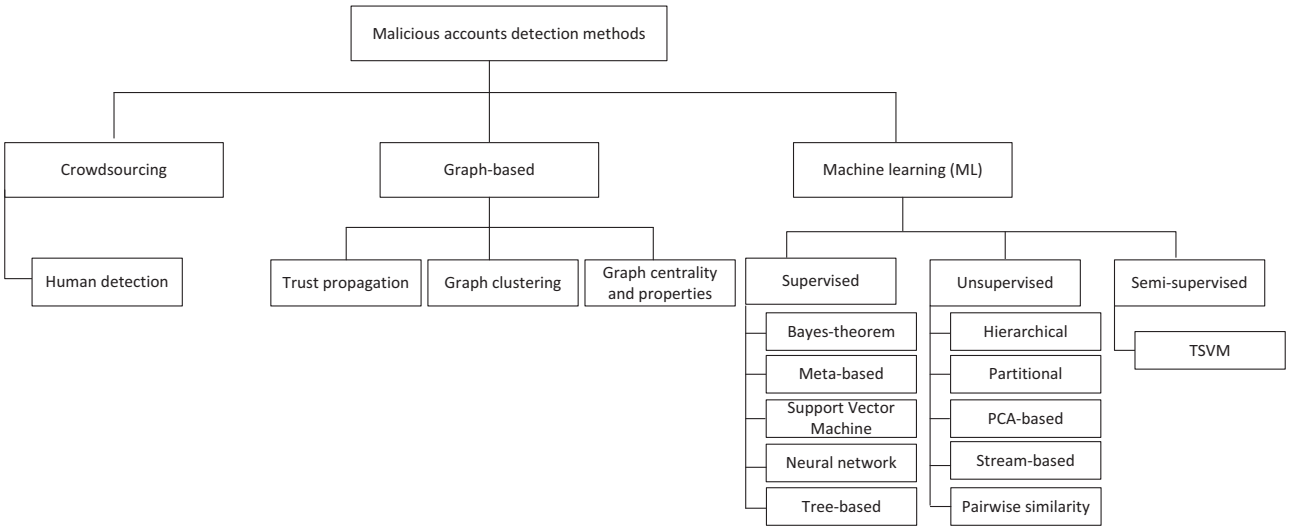

Fig. 7. Taxonomy of malicious account detection methods.

providing efficient detection models. However, by combining content and network information, the performance of their classification models improved. Similar findings have been reported on Renren network (Yang et al., 2014a) revealing that Sybil accounts on Renren collude together to spread similar contents. Wu et al. (2016) combined content and network information to develop a novel classification algorithm based on L1 and L2 regularization, which can identify spammers and spam message simultaneously. In a similar manner, Hu et al. (2013) developed a classification model to combine textual features with adjacency matrix represented by the users network connections. The main challenges with these unified analyses lie in the optimal performance of the classification systems, coupled with the need to identify the most discriminative features, which can be combined for better performance.

5. Taxonomy based on methods

This section presents a taxonomy of the different methods used in the literature to detect malicious accounts. Some researchers believe that a complete study of the social network structure is enough to identify malicious accounts, while others exploit feature-based approach as discussed in Section 4, utilizing machine learning algorithms to detect misbehaving accounts. In this paper, the taxonomy by method is referred to as the final approach applied to categorize the accounts under investigation. Fig. 7 shows the taxonomy of the different methods for detecting malicious accounts and their behaviors specifically in social networks. The methods include crowdsourcing, graph-based and machine learning (ML). The figure further shows that the crowdsourcing method uses human detection approach to identify malicious accounts. The graph-based method consists of a variety of approaches, such as trust propagation (Ghosh et al., 2012; Yang et al., 2015), graph clustering (Ahmed and Abulaish, 2012; Viswanath et al., 2011) as well as graph centrality metrics and properties (Sadan and Schwartz, 2011). Machine learning method employs supervised, unsupervised, and semi-supervised approaches. This taxonomy allows us to gain more insights on the current methods for detecting malicious accounts. It is important at this point to give examples for illustration purpose. For instance, the graph clustering method presented in (Liu et al., 2015; Viswanath et al., 2011), which is based on community detection, does not utilize machine learning method. The authors only studied the community structure of social network accounts in order to partition nodes (i.e accounts) using their community similarity. In the research work of (Miller et al., 2014), the researchers extracted content/behavioral features to train stream-based machine learning algorithms. Some studies that utilize machine learning method trained different machine learning algorithms to check the performance of their proposed features across varieties of classifiers. This is discussed in details under machine learning method.

Table 6 shows the studies that deal with malicious behavior detection on OSNs using the three aforementioned methods. The table provides a general overview of the current state-of-the-art literature in malicious accounts detection with a specific focus on social networks.

5.1. Crowdsourcing

Wang et al. (2012a) suggested the method of crowdsourcing for detecting malicious accounts. This method leverages human detection, which distributes intelligent tasks to Internet users who can identify a pattern of anomalies exhibit by social network accounts. Crowdsourcing involves the use of large and distributed group of workers known as crowd workers to identify suspicious behaviors. The crowd workers analyze social network accounts by checking the information on their profiles and decide whether the accounts are Sybil or legitimate. In crowdsourcing approach, a platform is developed for crowd workers, where they can assess users' profiles and make a decision based upon the outcome of their investigation. For instance, Tuenti, which is the largest social network in Spain employed 14 full-time employees to detect fake accounts on its network (Cao et al., 2012). By applying crowdsourcing method on two popular OSNs platforms: Facebook and Renren, Wang et al. (2012a) observed that the performance of the hired crowd workers reduces over time, although this method brings about a concession where majority votes can be used to reach the final judgment. This strategy is found suitable for social network providers since it demonstrates a near-zero false alarm. However, a number of drawbacks hinder the applicability of this method when used to detect malicious accounts.

First, Wang et al. (2012a) stated that crowdsourcing method is effective if adopted by the social network providers at the early stage. This shows that crowdsourcing will incur a high cost if use on social networks with a large number of pre-existing users, such as Facebook and Twitter. Virtually all social networks have recently reported a huge increase in their registered users and social interactions; therefore, adopting this method may incur a high cost for social networks with large users. Second, this method still requires the knowledge of experts to guarantee reliable annotation. However, not all crowd workers possess the expert knowledge needed to produce zero false alarm (Wang et al., 2012a). Third, exposing personal data of social network users to external crowd workers may raise the issue of privacy and this

Table 6
Summary of studies on malicious accounts detection.

Year	Ref.	Objectives	Database source	Detection category	Method
2006	Yu et al. (2006)	Proposed SybilGuard based on fast-mixing assumption and random walks	ACM	Fake account	Graph-based
2009	Markines et al. (2009)	Proposed framework for spam detection in social tagging system	ACM	Spam account	Machine learning
2009	Benevenuto et al. (2009)	Developed ML model to detect spammer on YouTube	ACM	Spam account	Machine learning
2010	Viswanath et al. (2011)	Analyzed Sybil defense schemes and developed a community-based Sybil detection approach	ACM	Fake account	Graph-based
2010	Gao et al. (2010)	Proposed a clustering approach to group spam into campaigns	ACM	Spam account	Graph-based
2010	Wang (2010a)	Applied profile-based and content-based features to detect spammer	Springer	Spam account	Machine learning
2010	Lee et al. (2010b)	Deployed honeypots and ML system to detect spammer on Twitter and MySpace	ACM	Spam account	Machine learning
2010	Stringhini et al. (2010)	Analyzed the impact of spamming on OSN and developed ML classifier to detect spammers	ACM	Spam account	Machine learning
2011	Sadan and Schwartz (2011)	Developed graph-based model using betweenness centrality metric	ScienceDirect	Phishing detection	Graph-based
2011	Tran et al. (2011)	Proposed a decentralized node admission control protocol algorithm called GateKeeper to separate Sybil accounts from normal accounts	IEEE	Fake account	Graph-based
2011	Yang et al. (2011)	Combined network and content-based features to detect spammer	Springer	Spam account	Machine learning
2011	Mccord and Chuah (2011)	Analyzed content features for spam accounts detection on Twitter	Springer	Spam account	Machine learning
2011	Kontaxis et al. (2011)	Developed tool to detect fake account on LinkedIn	IEEE	Fake account	Machine learning
2011	Jin et al. (2011)	Proposed framework to identify suspicious identities on Facebook	ACM	Fake account	Machine learning
2011	Stein et al. (2011)	Presented the underplaying design of Facebook Immune System	ACM	Fake account	Graph-based
2012	Wang et al. (2012b)	Analyzed clickstream data to detect existence of malicious crowdsourcing platforms	ACM	Fake account	Crowdsourcing
2012	Wang et al. (2012a)	Proposed crowdsourcing platform to detect fake accounts	ACM	Fake account	Crowdsourcing
2012	Ahmed and Abulaish (2012)	Applied MCL algorithm to cluster social network accounts into spam and non-spam	IEEE	Spam account	Graph-based
2012	Cao et al. (2012)	Developed SybilRank algorithm using power iteration approach	ACM	Fake account	Graph-based
2012	Ghosh et al. (2012)	Analyzed link farming activities used by accounts on Twitter	ACM	Spam account	Graph-based
2012	Conti et al. (2012)	Studied time evolution of social graph to detect fake accounts on social network	IEEE	Fake account	Graph-based
2012	Chu et al. (2012b)	Developed ML model to detect spam campaigns on Twitter	Springer	Spam account	Machine learning
2012	Aggarwal et al. (2012)	Proposed a tool called PhishAri for real-time detection of malicious tweet	IEEE	Phishing	Machine learning
2012	Chu et al. (2012a)	Focused more on automated account detection approach to identify malicious socialbots, human, and cyborg accounts	IEEE	Spam account	Machine learning
2012	Jiang et al. (2012)	Proposed Sybil group detector on Renren network	IEEE	Fake account	Machine learning
2012	Gani et al. (2012)	Proposed framework that relies on ML model, social interaction and authorship analysis for fake account detection	ACM	Fake account	Machine learning
2013	Lin et al. (2013)	Focused on introducing lightweight features for phishing detection	IEEE	Phishing	Machine learning
2013	Yang et al. (2013)	Combined network and content/behavioral analysis to detect spammers	IEEE	Spam account	Machine learning
2013	Tan et al. (2013)	Designed unsupervised Sybil defense scheme to identify spam accounts in OSN	ACM	Spam account	Graph-based
2013	Martinez-Romo and Araujo (2013)	Combined language model and tweet content approaches to detect spammer	ScienceDirect	Spam account	Machine learning
2013	Lin and Huang (2013)	Studied features for detecting long-surviving spammers on Twitter	IEEE	Spam account	Machine learning
2013	Ahmed and Abulaish (2013)	Proposed 14 generic features for spam detection on Twitter and Facebook	ScienceDirect	Spam account	Machine learning
2013	Bhat and Abulaish (2013)	Developed spam account detection system using community-based features	IEEE	Spam account	Machine learning
2013	Li et al. (2013)	Proposed semi-supervised approach to detect phishing attack	ScienceDirect	Phishing	Machine learning
2014	Chen et al. (2014)	Proposed different features for phishing detection on social network	ScienceDirect	Phishing	Machine learning
2014	Alsaleh et al. (2014)	Classified accounts on Twitter as human, bots, and Sybil using ML models	IEEE	Fake account	Machine learning
2014	Galán-García et al. (2014)	Detected spammers account used for cyberbullying on Twitter	Springer	Fake account	Machine learning
2014	Yang et al. (2014b)	Developed real-time Sybil detector on Renren	ACM	Fake account	Machine learning
2014	Chan et al. (2014)	Proposed re-weight method in adversarial learning for spam filtering in OSN	ScienceDirect	Spam account	Machine learning
2014	Bhat et al. (2014)	Trained ensemble of classifiers using community-based features	IEEE	Spam account	Machine learning
2014	Singh et al. (2014)	Developed ML model for malicious account detection on Twitter	ACM	Spam account	Machine learning
2014	Fire et al. (2014)	Developed social privacy protector system for fake account detection on Facebook	Springer	Fake account	Machine learning
2014	Lee and Kim (2014)	Developed model using name-based features to detect malicious account	ScienceDirect	Fake account	Machine learning
2014	Kiruthiga et al. (2014)	Introduced extended clone spotter algorithm that employed classification and clustering techniques	IEEE	Fake account	Machine learning
2014	Miller et al. (2014)	Modified stream clustering algorithms to detect spammers on Twitter	ScienceDirect	Spam account	Machine learning
2015	Yang et al. (2015)	Developed VoteTrust algorithm to detect Sybil accounts using signed graph	IEEE	Fake account	Graph-based
2015	Gupta and Kaushal (2015)	Combined different learning algorithms to detect spam accounts on Twitter	IEEE	Spam account	Machine learning
2015	Zheng et al. (2015)	Developed tool to detect Sina Weibo spammers	ScienceDirect	Spam account	Machine learning
2015	Egele et al. (2015)	Analyzed and proposed compromised accounts detection framework in OSNs	IEEE	Compromised account	Machine learning
2015	Cao and Caverlee (2015)	Proposed framework based on posting and clicking behaviors of posters and clickers of URLs to identify phishing links	Springer	Phishing	Machine learning
2015	Devineni et al. (2015)	Proposed PowerWall algorithm based on modified power law property of a social graph	ACM	Fake account	Graph-based
2015	Ezpeleta et al. (2015)`	Analyzed spam vulnerability with public profile information on OSN	Springer	Spam account	Crowdsourcing
2015	Cresci et al. (2015)	Introduced new baseline dataset for fake follower detection in OSN	ScienceDirect	Fake account	Machine learning

(continued on next page)

Table 6 (*continued*)

Year	Ref.	Objectives	Database source	Detection category	Method
2015	Liu et al. (2015)	Proposed community-based approach to identify social spammers based on two step-process	Springer	Spam account	Graph-based
2015	Main and Shekokhar (2015)	Proposed five features for spammer detection	ScienceDirect	Spam account	Machine learning
2016	Wu et al. (2016)	Proposed a unified framework based on network and content information	ScienceDirect	Spam account	Machine learning
2016	Igawa et al. (2016)	Developed a wavelet-based approach for account classification that detects textual dissemination of spam accounts	ScienceDirect	Spam account	Machine learning
2016	Ruan et al. (2016)	Introduced extroversive and introversive features based on clickstream to detect compromised accounts	IEEE	Compromised account	Machine learning
2016	Zhang and Lu (2016)	Proposed approach for detecting near-duplicate accounts on Weibo	Springer	Fake account	Graph-based
2016	Zuo et al. (2016)	Leveraged friends-of-friends relationship to detect misbehaving users	ScienceDirect	Spam account	Graph-based
2016	Mulamba et al. (2016)	Proposed SybilRadar, an algorithm that improves over SybilRank	Springer	Fake account	Graph-based
2016	Pérez-Rosés et al. (2016)	Studied endorsement relationship between accounts based on some selected skills	ScienceDirect	Spam account	Graph-based
2016	Almaatouq et al. (2016)	Applied Gaussian mixture model (GMM) to identified two categories of spammers and proposed network and content features	Springer	Spam account	Machine learning
2016	Harsule and Nighot (2016)	Developed a system called Filter Wall (FW) based on N-gram analysis	Springer	Spam account	Machine learning
2016	Cao et al. (2016)	Proposed forwarding-based and graph-based features for phishing detection	Springer	Phishing	Machine learning

can encourage even the crowd workers to exploit the concerned users (Wang et al., 2012b). Finally, several malicious crowdsourcing platforms are in existence, which negatively used their platforms to control a large number of accounts and make a huge financial gain (Wang et al., 2012b).

5.2. Graph-based

The possibility of modeling social network as a graph has played a key role in identifying malicious behavior in OSNs. A graph is formally represented as $G=(V, E)$, where V is a set of vertices and E is a set of edges (Al Hasan et al., 2006; Nettleton, 2013). In an online social network, this graph is referred to as a social graph. The interpretation of nodes and edges in the graph G varies according to the problem under consideration and the modeling technique. While an edge may represent friendship invitation (Yang et al., 2015), in some cases it may denote URL links between a pair of nodes (Tan et al., 2013). The increase in the growth of social graph is due to the addition of nodes and edges. The social graph can be unipartite, bipartite or tripartite. A unipartite social graph has one type of node, while a bipartite or tripartite social graph considered a graph with its nodes partitioned into multiple types (Savage et al., 2014; Vlasselaer et al., 2013).

The social graph can also be categorized into static, dynamic, labeled or unlabeled (Savage et al., 2014). Static network structure neglects time evolution of interactions among the individual nodes on the network. A dynamic network is a type of social network structure that changes over time, with the changes occurring as a result of patterns of interactions (Conti et al., 2012; Savage et al., 2014). A labeled network considers node attributes in addition to the nodes and edges on the network. In a social network, node attributes may represent name, age, sex, and organization. On the other hand, unlabeled graph structure only considers nodes and edges in the topology without node attributes.

Visualization of social graph especially reveals the so-called hubs, which are users with a large number of social links. Hubs have a great potential for interaction and communication on the networks (Heidemann et al., 2012) and they are mostly targeted by malicious users. For example, Fig. 8 shows Twitter graph visualization using label structure. Each node is labeled with the user screen name, which represents the node attribute. This attribute can be used to distinguish a node from another on the network. As seen from this graph, cshirky, located at the center of the graph, is an example of a hub with a large number of friendship connections.

To detect malicious accounts on the social graph, many graph-based methods have been studied. This section categorizes the methods into three: Trust propagation, graph clustering, and graph metrics and properties.

5.2.1. Trust propagation

Social graph can have two trust relationships: strong or weak trust. OSN graphs with strong trust are those that possess the property of fast-mixing (Mulamba et al., 2016; Yu et al., 2006). In fake accounts detection problem, this can be viewed as a social network with a small cut, which represents a set of edges that when remove will partition the graph into two regions of honest and Sybil. For the sake of clarity, OSN with strong trust relationships has a limited number of attack edges between honest and Sybil regions. Conversely, a social graph with weak trust does not possess the fast-mixing property. Another assumption similar to fast-mixing is the random expander assumption used for developing Gatekeeper algorithm (Tran et al., 2011). Mohaisen et al. (2010) demonstrated that many social networks are not fast-mixing, which indicates that the number of attack edges on several social networks can be in millions. Attack edges are the links between Sybil and non-Sybil regions. The link prediction problem can be used to predict such attack edges using feature similarity or social structural similarity (Mulamba et al., 2016). The former similarity measure

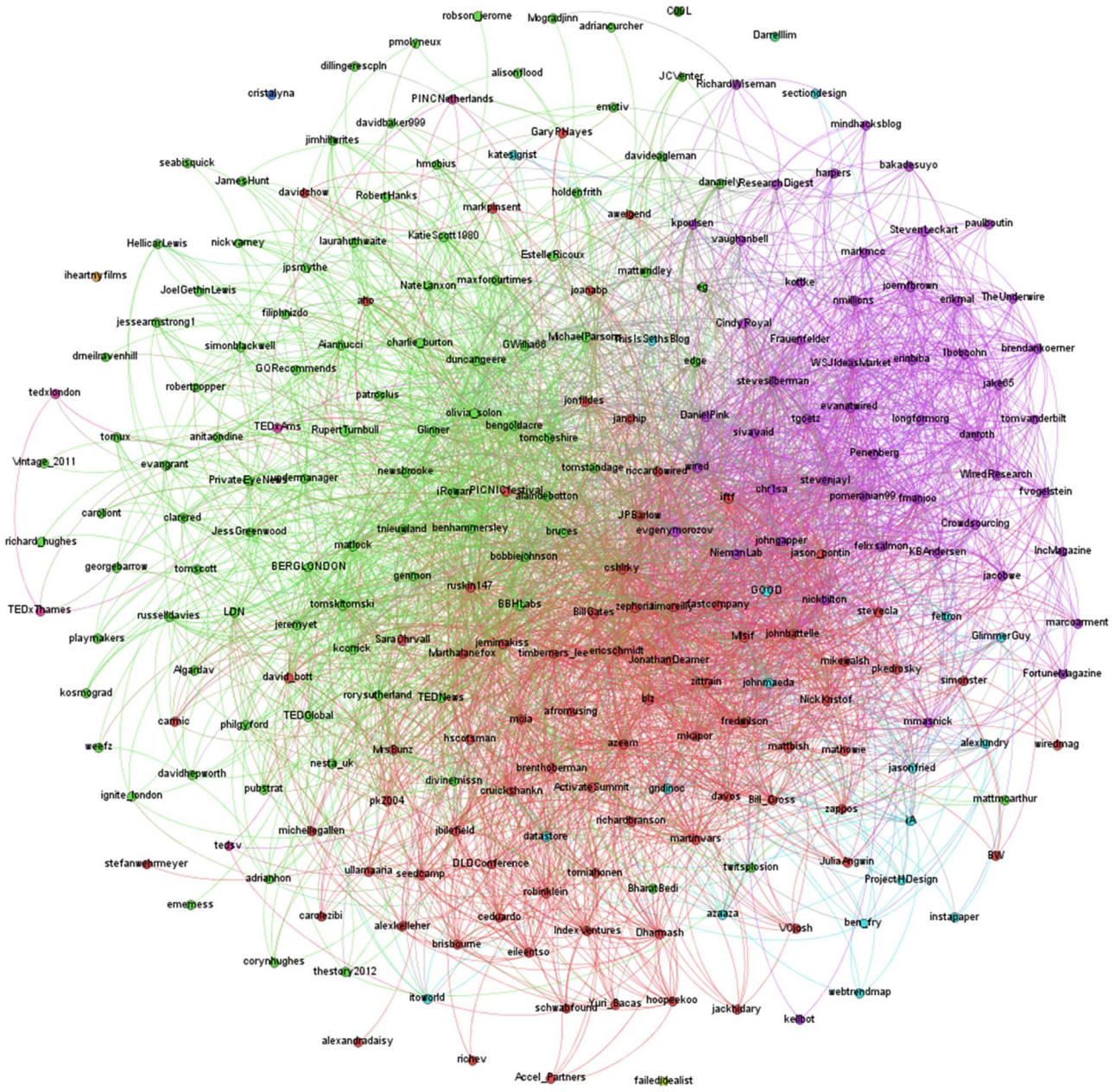

Fig. 8. Visualization of graph data using Gephi - an open-source software for visualizing and analyzing large network graphs.

considered the node attributes in the social graph while the latter only studies the structural link that exists between a pair of nodes. For instance, the similarity metric proposed by (Adamic and Adar, 2003) is one of the popular structural similarity metrics used to predict attack edges (Mulamba et al., 2016). Since the goal of fake account detection system using social graph method is to identify the misbehaving nodes, link prediction problem has been shown to perform poorly in a social network that exhibits weak trust relationship (Mulamba et al., 2016). Thus, with the use of trust propagation method, it is possible to improve detection of Sybil in OSNs.

In trust propagation method, a degree-normalized landing probability is computed and assigned to each node in the social graph. This probability corresponds to the probability of a modified random walk to land on each node. The random walk starts from a know non-Sybil node. This node distributes its trust value to the neighboring nodes. At each step of the random walk, a trust rank is computed, which indicates the strength of the trust connections that exist between the nodes. The step of random walk's probability distribution is a trust propagation process. It is important to note that a random walk can be made to terminate at an early stage; such random walk is called a short walk. A random walk that runs for a long period will produce uniform trust rank values for all the nodes in the social graph. This uniform trust value is known as the convergence value of the random walk. Random walk convergence relies on a number of steps known as the mixing time of the social graph (Cao et al., 2012; Mulamba et al., 2016). One of the popular algorithms for computing the trust value during the random walk is power iteration (Cao et al., 2012).

In the realm of Sybil detection on social networks, random walk approach has been widely used to separate Sybil from legitimate accounts. For instance algorithm such as SybilGuard (Yu et al., 2006), Gatekeeper (Tran et al., 2011), SybilLimit (Yu et al., 2008), SybilRank (Cao et al., 2012), and SybilRadar (Mulamba et al., 2016) used random walk technique to identify malicious nodes. Although the assumption made in SybilGuard, SybilLimit, SybilRank and Gatekeeper is quite different from the assumption used to develop SybilRadar. The latter Sybil detection algorithm assumed that the social network

exhibits weak trust due to the findings of different experiments, which indicate that Sybil account can establish large attack edges to legitimate accounts (Zhang and Lu, 2016). This makes SybilRadar detects a large number of Sybil nodes and outperforms the earlier algorithms for Sybil detection, such as SybilRank. As an example of how trust propagation is used, the process starts from the administrator identifying some trusted nodes as initial seeds for the algorithm. A random walk is then used to compute the trust value, which is the landing probability for each node on the graph. The trust values are sorted in descending order so that the higher rank nodes are put on the top (Cao et al., 2012). It has been shown that the early Sybil detection algorithm drops significantly in performance when the number of attack edges is increased (Mulamba et al., 2016; Viswanath et al., 2011; Yang et al., 2015). SybilRadar attempts to improve the performance of the early Sybil detection algorithms by introducing a number of stages based on social structural analysis to refine the performance of SybilRadar. However, as stated by the researchers, there is a limit to the application of only the social structure to effectively identify malicious accounts with temporal changes.

A variation to initial seeds selection using both legitimate and spammers accounts for trust rank computation was demonstrated by CollusionRank algorithm (Ghosh et al., 2012). The algorithm used both known spammers and legitimate accounts as initial seeds and assigned trust and untrust values to the neighbor of these accounts. The trust values are spread to all the accounts on the network. The value assigned to each account is used to depict the strength of trust and to identify other spammers on the network. However, this approach suffers from the setback of initially selecting the number of known spammers and legitimate accounts that can give a better representation of the entire accounts on the social network. Since the number of seeds is very limited taking into consideration the overall size of OSN. The initial score of the original seeds will quickly get diluted (Liu et al., 2015). This may propagate imprecise scores to many accounts on the network, which are not enough to categorize the unknown spammers or legitimate accounts.

Another algorithm (VoteTrust) that is based on trust propagation leveraged the friendship request acceptance between accounts in the social network (Yang et al., 2015). VoteTust algorithm proposed by (Yang et al., 2015) applied power iteration to compute the trust probability. VoteTrust is based on the rationale that a Sybil node can be detected by using the friendship request acceptance from a real user. A friend invitation between node pairs is then modeled as a directed signed graph, where an edge between two nodes takes the value of 1 or −1. A value of 1 on the edge indicates that the friendship request is accepted, while −1 indicates non-acceptance. Therefore, a node B is said to cast vote on node A, if B accepts or reject a request from A. One of the advantages of VoteTrust is that the algorithm exhibits high parallelism in processing large social graph. However, in some social networks (e.g Twitter), it is possible to launch an attack without necessarily befriending real users. This limits the capability of VoteTrust to detect some high-level malicious behavior.

5.2.2. Graph clustering

Social graph typically shows clustering characteristics. Graph clustering method attempts to group a set of related nodes on the graph based on their similarity. Two nodes are grouped only if they are within a specific distance to each other. The resulting groups from clustering are called clusters, which sometimes may be referred to as communities. The goal of the graph-based clustering algorithm is to group nodes into clusters by considering the edge structure of the graph in a way that increases edges within each cluster (Schaeffer, 2007). One of the widely used graphs clustering algorithms is Markov cluster (MCL). MCL accepts transition matrix from a weighted graph. By applying expansion and inflation operations, MCL iteratively clusters nodes on the graph and terminate once a stable matrix is obtained. The resulting clusters can be analyzed to detect malicious accounts (Ahmed and Abulaish, 2012). Ahmed and Abulaish (2012) extracted correlated information from user's profile, such as the URL shared, list of friends and Facebook fanpage-likes to generate a weighted matrix for the MCL algorithm. The result of the MCL clustering algorithm produces three clusters. The first cluster contains accounts classified as spam profiles, the second cluster contains accounts classified as normal profiles, and the third cluster contains accounts classified as both spam and normal profiles. The authors applied a majority vote technique to resolve the third cluster with overlapping classes.

Gao et al. (2010) proposed a clustering algorithm to group wall posts into spam campaigns. The model starts by representing wall post as a pair $<description, URL>$, where *URL* is the link embedded within the wall post and *description* is the content of the wall post. The process connects two wall posts together if they link to the same destination URL. This results in a wall post similarity graph. The connected subgraphs in the wall post similarity graph depict clusters. Applying two widely used properties for identifying spam campaign, distributed and bursty, each cluster classified as malicious or benign. The time complexity of this algorithm is $O(n^2)$, which limits its applicability to identify spam campaigns in the large social network graph. To address robustness against spam attack, UNIK (Tan et al., 2013) algorithm uses the assumption that the URL non-spam patterns should be identified since they exhibit a more relatively stable pattern than the spam URL. UNIK algorithm is robust to an increasing level of spam attack. However, UNIK suffers from shorten URL attack strategy and attacks coming from compromised accounts on the network.

Another domain of graph clustering focuses on the detection of communities that can capture the notation of malicious and legitimate accounts clusters. One of the assumptions about the real-world social network is that nodes in this network are densely connected to each other, but are sparsely connected to another dense group in the network (Bhat and Abulaish, 2013). The densely connected group of nodes is referred to as a community. Detecting community is an important step to identify malicious group, and to study the behavior of this group on the network. Using the idea of subgraph mining in graph theory, community detection algorithm attempts to cluster set of nodes that share common characteristics into community based on their connectivity. For instance, Viswanath et al. (2011) adapted Mislove (Mislove et al., 2010) local community detection algorithm to detect Sybil nodes in the social networks. This algorithm starts by selecting a node as a seed and then iteratively adding neighboring nodes until the strong community is obtained. The node that improves the normalized conductance is added at every iteration. Conductance is a metric for evaluating the quality of communities within the large social graph. It is important to note that there are other metrics that can be used to measure the quality of community structure, such as the metric introduced by (Newman and Girvan, 2004). The Mislove community detection algorithm adapted by Viswanath et al. (2011) terminates when there is no node that produces an increased in the normalized conductance. Viswanath et al. (2011) modified the algorithm to continue executing until all the nodes on the network are ranked. A lower ranking value indicates that the node is Sybil (i.e fake account). This approach shows better performance when compared with early Sybil detection algorithms, such as SybilGuard and SybilLimit.

Liu et al. (2015) also proposed a community-based method, which uses a two-step process. The first step clusters accounts into communities and the second step assigns a label to each account in the community based on the features exhibit by the accounts and the community. The higher the numbers of community that two accounts belong to the higher their similarity. However, among the most noticeable challenges of Mislove community detection algorithm is lack of scalability, as this algorithm takes $O(n^2)$ time complexity. In some cases, community detection rarely provides provable guarantees

in detecting malicious accounts (Cao et al., 2012).

5.2.3. Graph centrality and properties

Interesting properties of social graph, such as power law distribution (Xin-fang, 2013), scale-free topological structure, small-world as well as graph centralities assist in detecting malicious accounts in social networks (Sadan and Schwartz, 2011). Scale-free network is a network having degree distribution that follows a power law (Onnela et al., 2007). This means that the probability distribution of the number of connections between nodes in the network follows a power law distribution. This assumption also holds for clustering coefficient, the vertex connectivity between nodes, and small average path length (Sallaberry et al., 2013). Although some real-world social networks are assumed to be scale-free (e.g web graph and co-authorship), this assumption has not been generalized to all real-world social networks. As a result, the scale-free properties of many social networks is still being debated in the social network research community (Clauset et al., 2009). Graph centrality metric measures the relative importance of each node on the social graph based on position. A node with a high value is assumed to be more relevant. However, the definition of this relevancy depends on the application domain of the problem under consideration. Betweenness is a centrality metric that determines how often a node is located on the shortest path between other nodes in a social graph. The metric represents percentages of all shortest paths in a network that pass through a particular node (Sadan and Schwartz, 2011). Other centrality metrics used in graph theory include closeness, PageRank, and eigenvector centrality.

Betweenness centrality metric has been applied in phishing URL detection to reduce false alarm (Sadan and Schwartz, 2011). This approach was tested on ground-truth data of about 10,000 randomly selected domains from the URIBL blacklist and whitelist data source. A link graph was constructed for each selected domain and the betweenness centrality values computed. The study shows that the betweenness centrality value of whitelist domains is notably higher than the blacklist. The strength of this approach is that it provides a powerful metric and effective tool that can complement URL based anti-spam systems as well as a reduction in false positives (Sadan and Schwartz, 2011). As observed in (Devineni et al., 2015), one of the issues with traditional power laws assumption is the inability to explain deviating behaviors. An extension of power law distribution called PowerWall was used to analyze the Facebook wall activities of users in an attempt to capture misbehaving accounts. The authors analyzed users' wall posts and identified patterns of anomalies from accounts that post the same number of messages every two days and another user who post every night without other activities (Devineni et al., 2015).

5.3. Machine learning

Machine learning (ML) has played significant roles in identifying malicious accounts in social networks. In fact, the majority of the articles on malicious accounts detection focused on machine learning. ML incorporates a variety of methods, such as supervised, unsupervised, and semi-supervised learning. Supervised ML algorithm acquires a labeled dataset and learns a model as output, which can predict the class label for new data (Narudin et al., 2014). In supervised learning, the classifier learns from a large quantity of label data to build a model during training. Unsupervised learning (i.e clustering) differs in the sense that, no labeled data is present during the training stage, and the system learns from the data itself by identifying relationships or similarities among the instances in the dataset. Because the process of obtaining labeled data is tedious, a semi-supervised algorithm takes little labeled data in addition to a large amount of unlabeled data to produce a model.

5.3.1. Supervised learning

Supervised learning is ML task of inferring a function from labeled training instances that consists of a set of observed examples. In supervised ML, an individual example is a pair consisting of an input typically a vector and the desired output value. Supervised ML analyzed training data to produce a classification model for predicting unseen data. In this learning category, the example data is converted to a series of feature vectors that consist of a set of values for each attributes (Zheng et al., 2015). At the end of training, a classification model is used to distinguish malicious and legitimate accounts (Singh et al., 2014).

Since supervised ML relies on labeled data and set of features, obtaining unbiased labeled data is very challenging. Some studies utilized manual approach to label their data (Chu et al., 2012a; Martinez-Romo and Araujo, 2013), by observing some distinguished characteristics of malicious users, such as following large users with less followers, posting unsolicited message, posting duplicate contents, excessive redirections, posting contents that lack intelligence and so on. The use of blacklist APIs and the combination of the blacklist, honeypot, and a manual method for data labeling have been explored (Aggarwal et al., 2012; Chu et al., 2012b; Lee et al., 2010a; Yang et al., 2013). Researchers also relied on the accounts suspended by social network providers, such as in the case of Twitter suspension algorithm (Almaatouq et al., 2016; Thomas et al., 2011). Singh et al. (2014) applied automated approach for data labeling using an analytical model with the equation of two entities: user score and tweet score. This approach biased label data along four features of interest: the number of followers, the number of tweets, the number of following, and account creation date, and defined some thresholds to compare the two entities. Based on this comparison, the account is assigned a class label, which may represent malicious, non-malicious or celebrities. Although Singh et al. (2014) achieved an accuracy of 99.8% with this approach using a Random Forest classifier, however, the underlying assumption that surrounds their labeled data may cause an increase in false alarm when deployed in a real-world situation.

It is important to note that several studies that applied supervised ML method are motivated by the need to introduce new features. Once a set of features have been identified, the next is to train machine learning classifiers. For the purpose of clarity, Singh et al. (2014) trained five classification algorithms: BayesNet, NaiveBayes, Sequential minimal optimization (SMO), J48, and Random Forest using content/behavioral features. They observed that Random Forest outperformed other four classifiers with an accuracy of 99.8%. Almaatouq et al. (2016) trained six different classifiers: ZeroR, Bayesian network, NaiveBayes, logistic regression, decision trees, and Random Forest using both content/behavioral and network features. They also observed that the performance of decision tree and Random Forest have been quite interesting. Therefore, this section discusses different machine learning classifiers and their various categories to detect malicious users. The emphasis is particularly on the classifiers with better performance based on the metrics utilized to evaluate them.

a) Bayes-theorem

Bayes' theorem is a statistical theorem that describes the probability of a hypothesis based on some given conditions. The theorem provides a way to understand how the probability that a given hypothesis is true is affected by a set of evidence. Bayes theorem has applications in a wide variety of domains, ranging from topic modeling (Kharratzadeh et al., 2015) to spam filtering (Chu et al., 2012a) in social networks. NaiveBayes and Bayesian Network algorithms built on top of this theorem and they have shown good performance in spam account and malicious URL detection in social networks (Chen et al., 2014; Wang, 2010a; Yang et al., 2011).

For instance, Yang et al. (2011) combined network and content features to train a NaiveBayes classifier. The authors trained NaiveBayes classifier with 18 features ten (10) of which were newly introduced in the study. The NaiveBayes classifier achieved a

detection rate of 88.6% when manually evaluated on some samples identified as spammers. Almaatouq et al. (2016) also combined content/behavioral and network features to train NaiveBayes and Bayesian Network algorithms. They applied Gaussian mixture model (GMM) to identified two categories of spammers: compromised and fraudulent accounts. The resultant clusters generated by GMM were used to construct the follower relationship graph used in the analysis. The authors extracted features around the follower relationship and contents posted to train the Bayes algorithms in addition to other five classifiers selected in the study. Chen et al. (2014) applied Bayesian Network to evaluate the discriminative power of some URL-based features. The authors examined seven features based on traditional heuristics and social network attributes of malicious URLs. They investigated the combination of features that can produce an improved classification performance when trained with Bayesian classifier. However, their findings revealed that combining all the features in their dataset improves classification accuracy with Bayesian classifier. In addition, NaiveBayes has shown good performance accuracy in classifying Twitter accounts as human, hybrid or Sybil (Wang, 2010b). Cao et al. (2016) analyzed the forwarding patterns of malicious URL on Sina Weibo social network. They applied URL-based features to train three classifiers with Bayesian Network achieving the highest accuracy. The primary advantage of Bayes-theorem based approach is the speed in training the classifier, since, in most cases, a single pass over the entire dataset is sufficient. However, a notable weakness of this approach begins when the dataset contains a large number of features.

b) Meta-based

The meta-based classifier is a family of supervised learning algorithms aimed at improving the generalization ability of the learned models. Meta-based classifier has no implementation of a classification algorithm on its own; instead, it utilizes other classification algorithms to perform the actual task. In addition, meta-based learning attempts to predict the good classifier for a given task based on the nature of the dataset. Therefore, it helps user in choosing which algorithm is suitable to apply to a given problem (Pappa et al., 2014).

Lee et al. (2010a), Markines et al. (2009) reported the performance of this classification model on social network data. For instance, Decorate, a meta-learning algorithm for developing various ensembles of classifiers successfully detect spammers who interacted with the social honeypots deployed on Twitter and MySpace networks (Lee et al., 2010a). In this study, the authors extracted profile-based features from the accounts identified by the honeypots approach. They trained machine learning algorithms based on these features. Out of the ten (10) classification algorithms investigated, Decorate classifier produced the best result. Markines et al. (2009) proposed AdaBoost model to detect spam accounts in a social tagging system. This classifier outperformed LogitBoost and linear SVM with an error rate of 2%. Fire et al. (2014) developed a social privacy protector for Facebook users with Rotation Forest ensemble algorithm achieving the best accuracy among the seven (7) classifiers considered in the study. One of the major areas that have been investigated when developing ensemble approach is how to combine weak classifiers to produce a single strong one. Bhat et al. (2014) have shown that ensemble approach can significantly improve the performance of individual classifiers. The authors improve the performance of their proposed model using J48 algorithm as the meta classifier for the ensemble task. In this study, the ensemble algorithm was trained using the community-based structural features extracted from the social interactions of the users on the network. A social network accounts use to distribute spam may exhibit a high proportion of spam word in posting patterns. Detecting this proportion with other content features can help identify malicious account. Such posting pattern can be detected by combining different algorithms, such as NaiveBayes, clustering and decision tree (Gupta and Kaushal, 2015). This approach achieved high accuracy with non-spam account detection, however, the accuracy of spam accounts identified by this meta approach needs to be improved (accuracy is 87.9%). Another issue with ensemble approach is how to identify the independent classifiers that can produce optimal performance over the given dataset.

c) Support vector machine

With the intention of reducing the error rate in classification task, while maintaining high performance accuracy, support vector machine (SVM) is implemented to detect malicious accounts. SVM is a statistical supervised learning model that analyzes data and detects patterns using label samples. SVM was developed at AT & T Bell Laboratories by Vapnik and co-workers (Smola and Schölkopf, 2004). This method can be used for both classification and regression problems. The goal of SVM is to separate the boundary between different classes in a dataset by defining a separating plane called hyperplane. This hyperplane separates the classes by maximizing the margin among the closest points known as support vectors from each class to the hyperplane. In the case of a nonlinearly separable problem, SVM uses kernel functions to find an optimal separating hyperplane. Examples of kernel functions used by SVM include linear, Radial Basis Function (RBF), and polynomial kernel.

In the domain of malicious accounts detection, several models based on SVM algorithm have been developed (Galán-García et al., 2014; Lee and Kim, 2014). For instance, Lee and Kim (2014) trained SVM algorithm with different name-based features extracted from the agglomerative clustering stage. The result of the SVM classifier shows that the model can cluster distinguished account names and classify them as benign and suspicious in order to provide a fast filter on which in-depth analysis of potential malicious accounts can be conducted. With the use of authorship identification and SVM model, Galán-García et al. (2014) identified real users behind fake accounts used for cyberbullying attacks on Twitter. Benevenuto et al. (2009) proposed SVM classifier to identify spammers in video sharing networks (VSNs). They train SVM with three sets of attributes based on a total view of tag videos, total ratings of tag videos, and user rank features. Martinez-Romo and Araujo (2013) proposed a framework based on language model and tweet content to train SVM classifier and identify malicious tweets in a trending topic. The language model was extracted based on a set of tweets that are related to a trending topic, the suspicious tweet, as well as the page linked by the suspicious tweet. By combining language and contents features, the SVM classifier is able to produce an accuracy of 92.2% and FPR of 6.3.

As reported in the literature, SVM classifier demonstrates good performance in social spam filtering and malicious accounts identification. However, the issue with this approach lies in the training time when large social network data are to be trained. This is evident in the study conducted by (Zheng et al., 2015), where SVM took over one hour to build a model for detecting spammers on Sina Weibo social network.

d) Neural networks

The applicability of neural network for classifying social network accounts has also been investigated in some studies (Alsaleh et al., 2014; Igawa et al., 2016). Neural network has been used in various application domains, such as pattern recognition, disease diagnoses, image processing and speech processing. However, due to the high computational requirements of neural network, they have found little application in malicious accounts detection in social networks. Neural network, such as multilayer perceptron (MLP) has been used in the work of (Alsaleh et al., 2014). MLP is a class of feedforward artificial neural networks (ANN) that consists of activation units, usually referred to as artificial neurons and

weights (Noriega, 2005). MLP modifies the standard linear perceptron by including multiple layers, such as input, hidden, and output layers to solve both linear and non-linear classification problems. The algorithm maps input data to appropriate outputs. During the training stage, MLP applies a learning algorithm, mostly back-propagation, to adjust the weights so that the network can acquire the required knowledge to classify new unseen data.

Alsaleh et al. (2014) introduced a number of content/behavioral features extracted from tweet metadata. The authors trained MLP using gradient descent (GD) method with a learning rate of 0.3. In this study, 50 nodes of neurons were used in the hidden layer with a validation threshold of 20 and a sigmoid activation function. By applying GD as the training method for the MLP, the algorithm produced a detection rate of 88.57% based on three classes: human, hybrid, and Sybil accounts. However, in the case of a two-class problem, human and Sybil accounts, MLP achieved a detection rate of 95.09% with 0.0491 error rate. The authors observed that out of the four classifiers selected in their study, MLP achieved the highest detection accuracy. Igawa et al. (2016) trained MLP with newly introduced wavelet features. They observed that MLP with simple architecture produced good classification result when applied to distinguish human from automated accounts. Although MLP has shown better detection accuracies as reported in the literature, however, during the training stage, MLP took $O(w^3)$ where w is the number of weights. This computational complexity may be costly when applied in a social network environment.

e) Tree-based

Algorithms in this category exploit the power of decision tree, where a classifier is learned using a tree structure. In this tree, a node represents the test of an attribute value and a branch denotes the result of the test (Aggarwal et al., 2012; Yang et al., 2011). Decision tree algorithms, such as J48 (C4.5) and Random Forest have shown wide acceptance in the literature for identifying spam and phishing attacks on social networks. J48 decision tree is based on C4.5 algorithm, a decision tree algorithm introduced by Quinlan in 1993 (Quinlan, 2014). This algorithm is an extension of Iterative Dichotomiser 3 (ID3). C4.5 uses information gain to select the best attribute at each node of the tree. This attribute represents the best candidate to make a decision about the splitting of the tree. Conversely, Random Forest creates an ensemble of classifiers by constructing different decision trees using random feature selection and bagging approach at training time. The decision tree produces two types of nodes, such as the leaf node that is labeled as a class and the interior node associated with a feature. Different subset of training data is selected with a replacement to train each tree (Chu et al., 2012a; Narudin et al., 2014).

Random Forest algorithm has improved detection accuracy of spam accounts detection system (Igawa et al., 2016; Singh et al., 2014; Stringhini et al., 2010). For instance, Aggarwal et al. (2012) used Random Forest to identify malicious tweets on Twitter network. The authors trained Random Forest using four categories of features based on profile, URL, WHOIS, and tweet contents achieving an accuracy of 92.52% to distinguish phishing from safe links. In addition, Random Forest reduces false alarm and hence increases the precision on both the classes considered in this study. They further investigate the performance of Random Forest classifier by employing a confusion matrix, which shows that the classifier can detect 92.31% of phishing tweets correctly with misclassification error of 9.6%. However, it was discovered that the increase in misclassification of legitimate tweets as phishing was because the tweets were posted by users who exhibit close behavior to a phisher using excessive unrelated hashtags. To mitigate the rise in spammers activities on Twitter, Mccord and Chuah (2011) gathered 1000 accounts with their 100 most recent tweets. They extracted content/behavioral features to train four classifiers with Random Forest algorithm achieving the best accuracy (95.7%). To identify long-surviving spammers on Twitter, Lin and Huang (2013) applied J48 and validated the algorithm with 400 labeled Twitter accounts containing both 200 normal and spammers' accounts. With the intention of improving the performance of spam accounts detection system, Bhat and Abulaish (2013) extracted community-based features from a social graph. They trained different ML algorithms and observed that J48 algorithm achieved the best accuracy. Ahmed and Abulaish (2013) proposed 14 features with 7 identified as most discriminative. They trained different ML algorithms and achieve best classification performance with J48 decision tree.

While J48 and Random Forest improved classification performance in most of the cases reported, the J48 algorithm is prone to the overfitting problem. However, with the use of reduced error pruning method, the algorithm can achieve better performance when applied to classify unseen social network data. Random Forest attempt to correct the overfitting problem in most of the decision tree algorithms, but it faces the challenge of producing a user-friendly readable output for analysis.

5.3.2. Unsupervised Learning

Unlike supervised machine learning approach (i.e classification), unsupervised learning used unlabeled data to build a model. As such, no specific attack behavior is known apriori. The unsupervised method groups data into different classes according to their similar characteristics. The method attempts to learn from the data by observing the similarities among instances in the dataset. Unsupervised learning is very useful in pattern analysis and for grouping social spam into campaigns (Lee and Kim, 2014). The different unsupervised learning methods used in the literature is categorized into five groups: Hierarchical, Partitional, PCA-based, Stream-based, and Pairwise similarity.

a) Hierarchical

Hierarchical clustering (HC) groups data over a variety of scales using a tree structure. This tree is a multilevel hierarchy, where clusters at one level are merged or split to obtain clusters at the next level. HC is either bottom-up (i.e agglomerative) or top-down (i.e divisive). Agglomerative clustering builds hierarchy using bottom-up approach by assuming that each instance should initially form its own cluster. The algorithm then iteratively merges pairs of clusters as one move up the tree. Divisive type operates in the opposite way and assumes that all instances are initially in one cluster. The algorithm recursively splits the cluster as it goes down the tree (Kaufman and Rousseeuw, 2009).

Studies have shown that attackers collude to establish malicious group and control a large number of accounts on the network (Ahmed and Abulaish, 2012; Jiang et al., 2012). Jiang et al. (2012) developed an algorithm similar to agglomerative hierarchical clustering to detect Sybil group on Renren. The algorithm first identifies suspicious users using popularity and social degree property. Users on the suspicious list were merged into Sybil groups based on their IP address similarity. Lee and Kim (2014) applied agglomerative hierarchical clustering to cluster users on Twitter based on their account names. Initially, the algorithm creates *"n"* number of clusters with a single name object and then iteratively merges similar clusters into larger clusters using a similarity function. The algorithm converges after obtaining a single cluster or the termination condition is satisfied. To compare two names, the algorithm measures the likelihood that the names are generated from a Markov chain model. This novel approach detects malicious accounts at the time of creation without having to wait for the initiation of malicious behavior. One of its limitations is a lack of efficiency in providing a defense mechanism against an intelligent adversary who can launch complex attack strategy to generate valid account names on the network. One of the weaknesses of agglomerative clustering is that the algorithm does not scale well on large data. The algorithm takes time complexity of at least $O(n^2)$,

where n is the number of instances.

b) Partitional

Partitional clustering divides set of instances into non-overlapping clusters such that each instance is in exactly one cluster. K-means is an example of a prototype-based partitional clustering algorithm with many application areas (Gani et al., 2012; Li et al., 2012). K-means is a heuristics-clustering algorithm that clusters dataset into user-defined clusters K by minimizing the sum of squared distance in each cluster. In order to use K-means algorithm, there is a need to calculate the distance between a point to its centroid, for this reason, Euclidian distance is commonly used (Yang et al., 2015).

Gani et al. (2012) proposed a framework that relies on unsupervised ML model, social interaction, and authorship analysis to identify multiple fake accounts on Twitter. Using K-means and Kohonen map algorithms, they cluster multiple groups of similar identities and perform manual verification to identify fake accounts. Kiruthiga et al. (2014) introduced an extended clone spotter algorithm that leverage clustering technique to detect a group of fake accounts using K-means algorithm. During the execution of clone spotter algorithm, K-means redistribute the identified cluster from which the closest center distance is computed and update the mean of each cluster accordingly. The authors employed two similarity distance metrics: Cosine and Jaccard to find accounts with similar characteristics based on a set of features such as age, the number of visiting friends, the total number of friends, user click patterns, and user action time period. They further introduced an extension of clone spotter algorithm, which identifies real and fake profiles on Facebook network with improved results. Partitional clustering such as K-means algorithm is simple to implement and fast when clustering large dataset. This is due to the linear time complexity of K-means. However, it requires the prior knowledge of cluster number and sensitive to outliers, which may result in the inaccurate partition.

c) PCA-based

Principal component analysis (PCA) is a statistical tool for identifying patterns in high dimensional data. PCA is suitable for detecting variation in a dataset, suggesting that it is a good candidate for malicious behavior detection in social networks (Viswanath et al., 2014).

Motivated by the need to develop a malicious account detection system without relying on apriori knowledge of attackers' strategies, Viswanath et al. (2014) proposed a PCA-based detection system. The system captures normal user behavior within three to five principal components. Any behavior that deviates from this pattern is considered as anomalous. This approach identifies fake, compromised, and colluding Facebook users who are collaborating to boost their like activities. The authors established that the normal user behavior, such as Facebook pages liked by the user and the rate of like activity could be captured within a low-dimensional subspace amenable to the PCA algorithm. The PCA algorithm finds the latent features around the users like activities, which are enough to separate anomalous from legitimate behaviors. They project normal user behavior within the low-dimensional subspace and the residual subspace captures anomalies and noise in the data. To separate anomalies from noise, the authors compute a bound on the L2-norm in the residual subspace such that any data point whose L2-norm in the residual subspace exceeds the bound value is flagged as anomalous. While this approach is very promising toward identifying malicious behavior without relying on labeled data, the PCA algorithm implemented takes $O(n^3 + n^2m)$ time complexity during eigenvalue decomposition of the covariance matrix. Where "n" is the number of input dimensions and "m" is a total number of accounts considered. This computational complexity is on the high side when considering large data involved in social networks.

d) Stream-based

The basic idea behind the stream-based approach is motivated by the development of stream clustering algorithms to separate spam accounts from legitimate ones. Miller et al. (2014) adapted two stream-based clustering algorithms, DenStream, and StreamKM++ to detect spam accounts on Twitter. DenStream is a stream-based clustering algorithm that extends the traditional batch learning DBSCAN algorithm by defining core-micro-clusters rather than the core objects concept used in DBSCAN (Cao et al., 2006).

One of the problems with K-means algorithm is that it requires a predefined number of clusters K and random selection of initial centroid. To address the problem of random selection of initial centroid, Arthur and Vassilvitskii (2007) developed K-means++ algorithm, which selects initial point using uniform probability. Because K-means++ is designed to process batch data, the algorithm is inefficient when processing date that evolves continuously. StreamKM++ algorithm extends the K-means++ with the use of a weighted point (i.e coreset) to address the streaming data.

Miller et al. (2014) applied content features to train DenStream and StreamKM++ and achieve good performance accuracy. This approach treats spam account detection as anomaly problem, and with the use of labeled training data divided into 1500 normal and 100 spam, the algorithms separate malicious accounts from legitimate ones achieving 2.2% false alarm. The key advantage of the proposed variant is that it is good for detecting malicious users in an evolving social network data environment. However, this approach needs to be improved on a large dataset to ascertain it scalability in categorizing spammers from legitimate users. Data stream clustering introduced a number of challenges, such as the need to significantly reduce the stored data while still maintaining good clustering results, incremental clustering over time and how to adequately update changes in the existing clusters.

e) Pairwise similarity

Pairwise similarity is the method of comparing two accounts based on their activities to determine which account exhibit sudden malicious characteristics. By building a profile of legitimate behavior, it is possible to compare this behavior with incoming user's activities to ascertain whether the new user's behavior conform to the initial profile (Kontaxis et al., 2011). This method is effective for identifying anomalies in social networks. For instance, Ruan et al. (2016), studied the social behavior of users in OSNs to detect compromised account. To determine if a specific account is compromised, the authors studied the behavioral history of the legitimate owner over a specific period. They explored the clickstream activities using both extroversive and introversive user's social behavioral patterns to build effective behavioral model. This approach starts by applying Euclidean distance to measure the differences between two profiles. Given two profiles P and Q, which contains both extroversive and introversive feature vectors for each profile. Let $A = (a1, a2, ..., a_n)$ and $B = (b1, b2, ..., b_n)$ denote a feature vector for both P and Q. Euclidean distance between vector A and B is calculated as shown in Eq. (1) and Eq. (2) shows the computation of Euclidean norm between profiles P and Q based on the Euclidean distance for each feature vector. The higher the value of $Dist$, the more significant the two profiles differ. In Eq. (2), m denotes the number of features vectors. The authors considered eight extroversive and introversive behaviors. They further defined the concept of self-variance based on the mean differences between the pair of profiles as well as the standard deviation of the self-variance to refine the distance metric. Based on the self-variance and standard deviation, the behavioral differences between two profiles can be determined to detect if a profile is compromised.

$$E(A, B) = \sqrt{\sum_{k=1}^{n} (a_k - b_k)^2} \quad (1)$$

$$Dist(P, Q) = \sqrt{\sum_{i=1}^{m} (E_i)^2} \quad (2)$$

Egele et al. (2015) also developed a behavioral based model using pairwise similarity method to identify compromised accounts on Facebook and Twitter. The authors extract content features from user's messages to build a user's normal behavioral profile. Any significant deviation from this behavioral profile is considered as a form of anomaly and can be used to identify compromised accounts. Using message features, such as time sent, message source, message text, message topic, link in the message, direct user interaction, and proximity, a global thresholding value is computed that combined all the feature models. This global threshold is used to determine if a profile is compromised or not. The threshold indicates the percentage of violation of the normal user behavioral profile. Kontaxis et al. (2011) defined a similarity score based on common fields between a pair of profiles to detect fake accounts on LinkedIn network. Jin et al. (2011) proposed two statistical similarity measures using attribute and friend network similarity to cluster fake accounts on Facebook.

While this approach is promising towards identifying behavioral violation, however, the definition of what constitute normal user behavior is complex in the real world, especially on the social network with a diverse set of functions, such as Facebook. A slight deviation in normal user activities may create a problem for a model that relies only on pairwise similarity. As an evidence of this limitation, Egele et al. (2015) confirmed that an adversary can break their similarity measure by sending messages to evade detection. An approach proposed in the work of (Jin et al., 2011; Kontaxis et al., 2011) relies on exact matching of fields before detecting similar identities. Therefore, it is important to fine-tune models based on pairwise similarity in order to reduce the increase in a false alarm.

5.3.3. Semi-supervised learning

Semi-supervised learning algorithm attempts to identify a suitable classification model by combining both labeled and unlabeled data. Because of the difficulty in obtaining labeled data in most application domains, such as in the case of social networks, the semi-supervised algorithm tries to learn a suitable model by permitting a small quantity of labeled data with a large amount of unlabeled data. (Kondratovich et al., 2013; Li et al., 2013) demonstrated the applicability of this learning approach. Some popular semi-supervised learning algorithms include expectation maximization, self-training, transductive support vector machines (TSVM), and co-training (Zhu and Goldberg, 2009). Li et al. (2013) applied TSVM algorithm to detect phishing attack. They used both image and document object model (DOM) features to train TSVM algorithm. The authors introduced quantum-inspired evolutionary algorithm (QEA) to deal with the local convergence problem of TSVM. In this study, TSVM outperformed its SVM counterpart with a significant accuracy of 95.5%. Previous experiments have demonstrated that TSVM can improve in performance over its SVM counterpart by exploring both labeled and unlabeled data (Kondratovich et al., 2013). TSVM combines the regularization effectiveness of SVM with a direct implementation of clustering assumption. This approach is very promising and can be very useful in social networks where there is a scarcity of public labeled data to detect malicious accounts. However, TSVM suffers from a number of drawbacks, such as its difficult non-convex optimization problem and the need to estimate the ratio of positive or negative examples from the dataset. Several research efforts have been committed to improving TSVM algorithm in this direction (Singla et al., 2014; Zhang et al., 2009). Thus, a version of this promising approach will reduce the difficulty in labeled data collection for anomalies detection in social networks.

5.4. Methods comparison

This section compares the different methods discussed in the previous section as shown in Table 7. The comparison is based on some parameters, such as application, dataset, learning type, labeled sample, parameter settings, strengths, and weaknesses of each method. The application summarizes the applicability of the methods for malicious account detection in social network. For instance, some methods such as trust propagation, graph clustering and meta-based are widely used in the literature for malicious account detection while others have limited application. The dataset indicates the type of data provided to the method. Some methods support real-time and near real-time while others focus on batch learning approach. In addition, some of the methods require labeled data while others relax the requirement. The table further highlights the parameter tuning, as well as the strengths and weaknesses of the methods. For example, one of the advantages of human detection approach is that it allows crowd workers to make a collective decision on the account under investigation, which eventually produces moderate accuracy. However, this method is costly, time-consuming, and lacks scalability as earlier discussed. It is important to note that this table summarizes our findings on the various methods specifically in the domain of malicious account detection in social networks.

6. Open challenges

Numerous studies have addressed the problem of detecting malicious accounts and their activities in online social networks. Sections 4 and 5 have been dedicated to providing detail analyses of the different features and methods that have been utilized to detect misbehaving users. These sections also highlighted the strengths and weaknesses of each feature category and method. However, as social networks continue to grow the number of attacks keeps increasing and attackers continue to change their behaviors to avoid detection by the existing systems. Therefore, the task of detecting the authenticity of a user's account or activities in social networks is very challenging. This section discusses a number of challenges, which emerged during the course of this review.

6.1. Dataset

The most prominent challenge in detecting malicious accounts in social networks is the lack of public datasets. This is due to the fear of violating user's privacy and the huge efforts involved in annotating social network data. In some cases, researchers go to the extent of paying commercial services to acquire data for research purpose some of which were not delivered even after payment (Viswanath et al., 2014). Unavailability of public ground-truth data is really a challenge that hinders effective models benchmarking. The majority of the studies reviewed in this paper utilized different private datasets indicating that an unbiased models comparison will be difficult due to many conditions, such as dataset size, the number of features considered, the ground-truth quality, data crawling process, the type of method adopted, and so on. Although the goal of each study is toward identifying patterns that can capture most of the malicious accounts, thus, comparing two studies based on performance metrics demands repeating the experiments in the previous studies on the new dataset.

Another issue regarding the availability of dataset for research purpose is the policy introduced by some social network providers, which prevent researchers from sharing social network data. This means that each researcher needs a crawler to collect data from a social network of interest. Collecting data using a crawler is challenging as this may introduce a delay in data collection as well as many noisy information. Most social network APIs allows users to execute a certain number of calls within a particular rate limit per hour. For instance, Twitter REST API allows 180 requests per 15-min window for user authentication and 450 requests per 15-min window for apps authentication (Twitter rate limit, 2015) using "search/tweets" request. To collect data from the Facebook network, Ruan et al. (2016) developed a

Table 7
Comparison of the different methods.

Methods	Application	Dataset	Learning type	Labeled sample	Parameter settings/ tuning	Strengths	Weaknesses
Human detection	Limited	Non-graph	Batch/Real-time	Not applicable	Not applicable	-Job creation -Allow majority vote -Moderate accuracy	-Costly -Time consuming -Lack of scalability
Trust propagation	Widely used	Graph	Batch	Seed selection	Minimal	-Performs user ranking easily -Good for modeling trust and distrust -Work well with dynamic network	-Sensitive to seed selection -Evasion -Rely much on assumptions -Communication overhead
Graph clustering	Widely used	Graph	Batch	Not applicable	Minimal	-Detects user community easily -Less assumption is made on the cluster statistics -Often more efficient than traditional clustering e.g K-means	-Scalability depends on algorithm. In most cases, less scalable on large network data - Highly sensitive to choice of parameters
Graph centrality and properties	Limited	Graph	Batch	Not applicable	Minimal	-Good for identifying node importance -Can be used to complement other methods	-Sensitive to neighborhood changes -Computationally expensive for large network data
Bayes-theorem	Widely used	Non-graph	Batch	Required	Less complex	-Fast training and prediction time -Good interpretable results	- Perform less on dataset with large features -Posterior distribution can be heavily affected by the prior information
Meta-based	Widely used	Non-graph	Batch	Required	Less complex	-Improvement in predictive accuracy -Availability of several approaches	-Identification of independent classifiers -Learning time and memory constraints
Support vector machine	Widely used	Non-graph	Batch	Required	Less complex	-Good accuracy from kernel choice -Better generalization	-Sensitive to choice of kernel and parameters -Training and testing speed is affected by data size
Neural network	Limited	Non-graph	Batch	Required	Complex	-Improved performance accuracy -Fast prediction time	-Slow training time -Performs less with small labeled samples
Tree-based	Widely used	Non-graph	Batch	Required	Less complex	-Better performance accuracy -Easily discover non-linear relationship -Rule generation	-Complexity of the trees and the information provided -Overfitting issue
Hierarchical	Limited	Non-graph	Batch	Not applicable	Minimal	-Good clustering quality -No apriori information of clusters number	-Computational complexity is high on large data -Often sensitive to noise and outliers
Partitional	Limited	Non-graph	Near real-time	Not applicable	Minimal	-Fast on large data samples -Simple to implement	-Selection of clusters number -Sensitive to outliers -Sensitive to initial seed selection
PCA-based	Limited	Non-graph	Batch	Not applicable	Minimal	-Provides good way of discovering latent features -Reduction in noise through maximum variability	-Computational complexity on large data is high
Stream-based	Limited	Non-graph	Real-time	Not applicable	Less complex	-Real-time detection -Scalable on large data samples	-Often produce clusters with low quality -In the case of DenStream, removal of outliers is a time-

(continued on next page)

Table 7 *(continued)*

Methods	Application	Dataset	Learning type	Labeled sample	Parameter settings/ tuning	Strengths	Weaknesses
						-Improved accuracy	consuming process
Pairwise similarity	Limited	Non-graph	Near real-time	Not applicable	Minimal	-Easy to implement -Good for modeling behavioral changes	-Evasion of similarity metrics -Complexity in pairwise comparison of users profiles
Transductive support vector machine	Limited	Non-graph	Batch	Required	Less complex	-Require small labeled samples -Better accuracy	-Local minimal issue -Computational complexity -Non-convex optimization

browser extension to log users' activities in the form of clickstreams. They recruited 50 Facebook users and collected their activities logs before the researchers could develop a compromised accounts detection system. This dataset collection process may be costly and may not provide an adequate method that can represent the entire users' population on Facebook network.

6.2. Features and evasion tactics

Research in machine learning has evolved through the introduction of many features to counter the new attack strategies posed by malicious users. Attackers have evaded some of these features using strategies such as posting heterogeneous messages to evade system that relied on message similarity, mixing good word with spam word, posting spam contents using automated tools, buying fake followers to grow their network, colluding with other malicious accounts to form a community, and so on. For example, to evade features, such as account reputation and follower/following ratios, attackers can buy more fake followers from the underground markets using the commercial services or exchange followers within their colluding networks (see Fig. 9 for pricing). Even the normal social network users are indirectly integrating malicious accounts in their network. For instance, consider a legitimate user who wants to boost his account reputation by purchasing fake followers from the underground markets. Unknown to the user that most of the fake accounts purchased hijacked from legitimate users for spamming activities or artificially created for malicious intents (Zhang and Lu, 2016). In fact, the so-called celebrities, politicians, and popular brands have purchased fake accounts from the underground markets to boost their profiles (Cresci et al., 2015). This makes detection of malicious account by content/behavioral based features much more challenging. It has been shown that structural based approaches that modeled social network as a graph with nodes and edges depicting user accounts and social connections can be evaded by an adversary who succeeded in establishing a large number of edges between the fake accounts and the legitimate users (Mulamba et al., 2016). Virtually any fake information can be purchased from the underground markets including fake likes, tags, ratings, comments, and accounts. This type of fake information undermined the essence of social network trust making it difficult to rely on the contents from the social media platforms.

To detect malicious accounts and their behaviors in social networks using content/behavioral approach, there is a need to observe the underlying detection system through an appropriate set of features and determine the features that provide good separation between malicious and legitimate activities. Considering appropriate features to use for improving malicious accounts detection system remains an open research issue. Although some studies have investigated the evasion tactics on both content and network features (Yang et al., 2013, 2011), however, to date no conclusion has been reached regarding the most discriminative features for malicious account detection. Yang et al. (2013) established that features, such as follower-following ratio, reputation, account age, bidirectional links ratio, betweenness, clustering coefficient, API ratio, API URL ratio, and API tweet similarity are good for detecting spammers accounts. Some researchers believe that by studying the network position of malicious accounts with the use of discriminative network features can help identify misbehaving users (Almaatouq et al., 2016; Cresci et al., 2015). However, more experiments are needed to verify these claims. In addition, some of the salient features are computationally expensive to extract from large social network.

6.3. Methods

Presently, there is a lack of universally acceptable method for malicious accounts detection in social networks. The crowdsourcing approach proposed in the literature is good for obtaining ground-truth

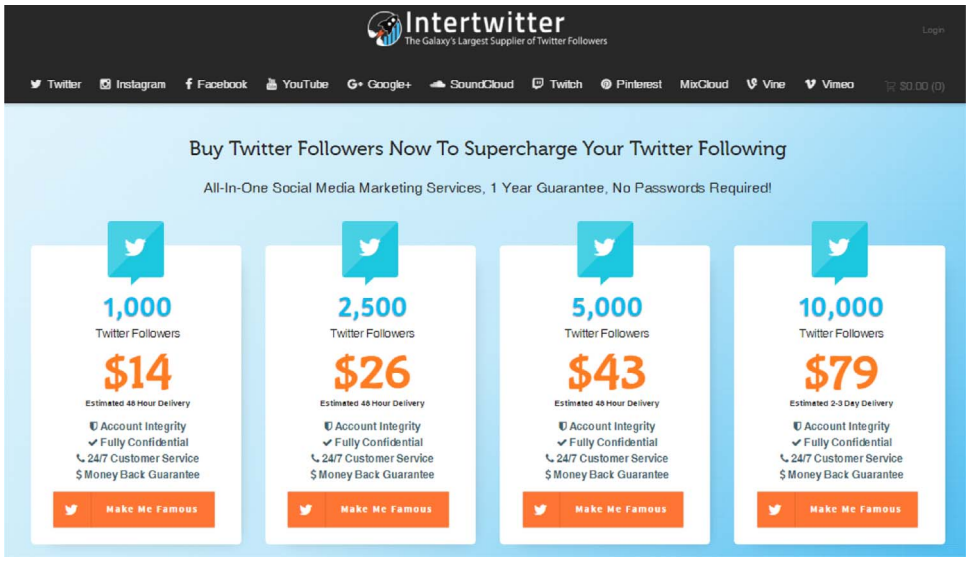

Fig. 9. : Price of fake Twitter followers from www.intertwitter.com.

data, mining OSN with a small number of users, and for creating a job opportunity. However, this approach suffers from many challenges including time wastage in data annotation and lack of scalability when investigating large social network users. In most cases, it involves spending a huge amount of money on the crowd workers to perform the human intelligence tasks. There is also the possibility of introducing human errors in data annotation. In addition, this method may give room for attackers to participate in the crowdsourcing tasks, which may create loopholes for vulnerability.

Graph-based method or structural method attempts to study the network structure using nodes and edges to uncover malicious activities. This approach is seen as being biased towards the structure of the graph under consideration and in some cases; they are mainly based on assumptions. In addition, such system is unable to detect intelligent adversaries who consciously change their behavior to evade detection. Most graph-based Sybil detection algorithms based their assumption on the fact that malicious users may find it difficult to connect with a large number of legitimate users in social network. However, the crucial issue is that even the legitimate users have started buying fake accounts and indirectly adding malicious users to their networks. With this in place, Sybil detection systems based on this assumption may end up misclassifying accounts on the social graph. A Sybil detection system that deviates from this assumption used friend invitation graph and assumed that attackers need to connect with legitimate users to launch attacks on the social network. However, in some social networks, it is possible to distribute malicious contents without establishing friendship connection.

Since malicious accounts detection problem can be viewed as a classification task, numerous studies conducted in the domain of machine learning mostly focused on supervised learning. Supervised methods require label data and are unable to detect zero-day malicious behavior. One of the challenges with this approach is that classifiers must learn from a good proportion of labeled data. It has been found that the proportion of legitimate accounts always surpassed the malicious ones. Therefore, there is a need for an efficient method to handle class imbalance and the need for a method that can auto-update the classification models when new labeled data are available.

6.4. Data streaming

Another challenge the existing systems for detecting malicious accounts in the social network may face is the speed at which social network data is evolving. In Big Data technology, this is referred to as velocity (Bhattacharya et al., 2016; Hashem et al., 2015). Data stream (Diallo et al., 2012) contains a possibly infinite sequence of data evolving at the different time. While existing studies have focused more on batch processing approaches, little research addressed the streaming nature of social network data (Miller et al., 2014). However, this study needs to be validated on large social network data to verify the scalability of the proposed methods. In addition, a distributed version of this approach can be incorporated in the future research to address potentially unbounded social network data. This can provide a scalable platform for real-time detection of malicious activities on the social networks.

6.5. Distributed implementation

Existing studies with the exception of (Yang et al., 2015) failed to address the growing nature of social network data on a large scale basis. Recently, the use of machine learning in distributed environments for processing large volumes of data is a recent research area. Some distributed data platforms recently introduced to address the problem of mining large dataset. This includes an open source project Apache Mahout and Apache Giraph implemented to work on top of Hadoop distributed processing framework. Mahout applies machine learning on top of Hadoop platform. Giraph is a graph processing platform that addresses scalability in graph mining research. The purpose of introducing Hadoop is to provide an enabling environment and programming models for distributed processing of large datasets across different clusters (Sethia and Karlapalem, 2011). Hadoop is made up of two primary components: Hadoop Distributed File System (HDFS) and MapReduce (Aridhi et al., 2015), which are closely related to each other. HDFS is a highly fault-tolerant distributed files system for storing data on Hadoop clusters. MapReduce is a high-performance programming model for distributed data processing (Hashem et al., 2015). These platforms can provide more insight on the process of mining large social network data for malicious behavior detection.

7. Discussion and future direction

As discussed earlier in Section 6, there are several challenges identified during the course of this review, which include dataset, feature evasion, methods, data streaming, and distributed implementation. To address some of these challenges, this paper proposes a unified framework in Fig. 10.

As an overview of the proposed framework, the process starts from

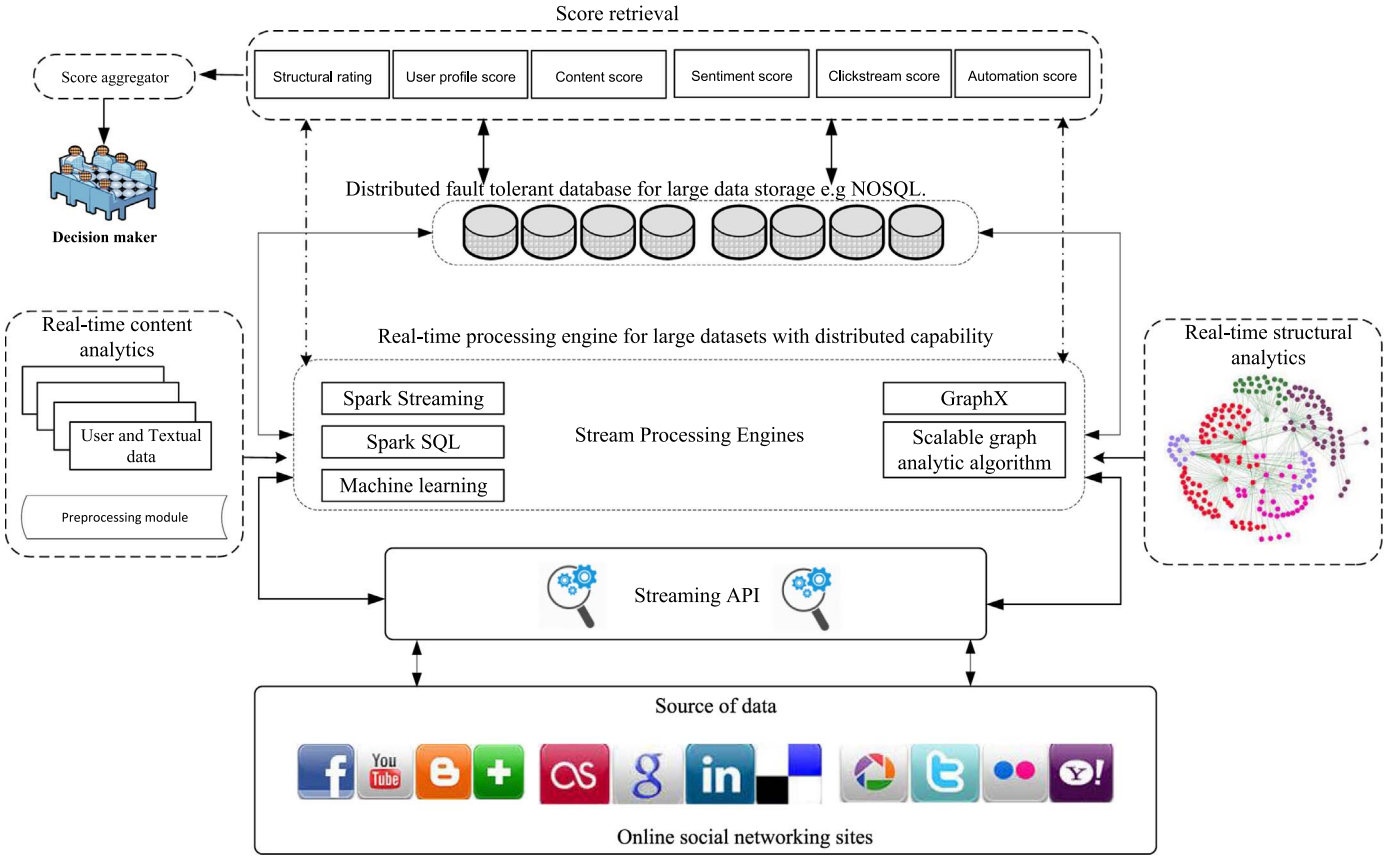

Fig. 10. Proposed framework for malicious account detection in OSNs.

data collection from social network of interest. In this case, a real-time data collection paradigm is proposed using the streaming API to get data in and out of the social media. Data streaming method allows for continuous data accessibility, which provides a platform for real-time data collection. The data collected are passed to real-time user and content analytic component that is embedded with the preprocessing module. The purpose of this component is to extract relevant features from user metadata and different messages posted on the network. This component builds efficient user profile by leveraging the scalable real-time processing engines, such as Spark architecture. Spark is an open-source solution for big data analytics, developed around speed, ease of use, and sophisticated processing (Apache, 2016). Spark provides support for scalable machine learning platform, graph analytics and streaming engine with cluster computing system that is compatible with Apache Hadoop. The machine learning module can be used to learn the user behavioral patterns around the metadata and textual contents extracted. This approach will produce user's rating, such as user profile, content, sentiment, clickstream, and automation scores. The real-time structural analytics component exploits the scalable platform for graph mining that is embedded with graph analytic algorithm to produce a score based on the user's social connection. Since the proposed framework takes advantage of both content/behavioral analysis and network information, there is a possibility of providing a platform that withstands attacker's evasion through a unified feature learning approach. The results of the real-time analytics processes are passed to the scalable fault-tolerant database and may be used directly by the decision maker, such as the case of spam and phishing detection system.

To detect fake and compromised accounts, the score retrieval component needs to first extract the previous user's history together with the current ratings and then pass the scores to the aggregator module. The aggregator module processes the scores and generates a unique value based on the user's behavioral scores submitted for evaluation. This value is sent to the decision maker, which in turn applies intelligent learning approach, such as fuzzy decision making to determine the class the user belongs. Note that this framework is generic and abstracts the core functionalities of each module. By combining content/behavioral and network information with real-time scalable data analytics, an improved malicious account detection system in social network is feasible.

8. Conclusion

The continuous increase in the volume of social network data has contributed to the growth in malicious activities. These days, social network users spend a significant amount of time storing and sharing useful information. This information has drawn the interest of cyber-criminals who exploit the trust relationship among social network users to carry out large-scale malicious campaigns. To reduce the effect of these malicious activities in social networks, researchers have proposed different features and methods to identify malicious users and their behaviors.

This paper reviewed a number of articles that deal with malicious accounts detection in social networks with a specific focus on spam accounts, fake accounts, compromised accounts, and phishing detection. To provide an effective way of categorizing these articles, the paper proposed taxonomies based on features and methods. The paper identifies issues and challenges with existing features and methods, which have been studied to detect misbehaving users and their activities.

The findings reveal that developing system for malicious accounts detection in social networks has been quite challenging. This is due to the evasion tactics posed by malicious users. However, there are rooms for improving the existing system for malicious accounts detection in social networks, such as identifying features based on content/behavioral and network information to capture a large number of malicious

accounts and their behaviors, as well as the need to address model scalability. A model for malicious accounts detection should address the rate at which social network data is growing on a daily basis and the speed at which the data is evolving. Highly efficient methods are needed to extract salient features from social network data. In addition, there is a need for better approaches to mine large social network graph for malicious accounts detection. Therefore, this paper predicts scalable malicious accounts detection system as an important area for future research.

Acknowledgement

The work of the authors is supported by University Malaya Research Grant Programme (Equitable Society) under grant RP032B-16SBS.

References

Ab Razak, M.F., Anuar, N.B., Salleh, R., Firdaus, A., 2016. The rise of "malware": bibliometric analysis of malware study. J. Netw. Comput. Appl. 75, 58–76.
Adamic, L., Adar, E., 2005. How to search a social network. Soc. Netw. 27 (3), 187–203.
Adamic, L.A., Adar, E., 2003. Friends and neighbors on the web. Soc. Netw. 25 (3), 211–230.
Aggarwal, A., Rajadesingan, A., Kumaraguru, P., 2012. PhishAri: Automatic Realtime Phishing Detection on Twitter. eCrime Researchers Summit (eCrime), 2012.
Ahmad, F., Sarkar, A., 2016. Analysis of dynamic web services: Towards efficient Discovery in cloud. Malays. J. Comput. Sci. 29 (3).
Ahmed, F., Abulaish, M., 2012. *An MCL-based approach for spam profile detection in online social networks*. In: 2012 IEEE Proceedings of the 11th International Conference on Trust, Security and Privacy in Computing and Communications(TrustCom).
Ahmed, F., Abulaish, M., 2013. A generic statistical approach for spam detection in online social networks. Comput. Commun. 36, 1120–1129. http://dx.doi.org/10.1016/j.comcom.2013.04.004.
Akoglu, L., Tong, H., Koutra, D., 2015. Graph based anomaly detection and description: a survey. Data Min. Knowl. Discov. 29 (3), 626–688. http://dx.doi.org/10.1007/s10618-014-0365-y.
Al Hasan, M., Chaoji, V., Salem, S., Zaki, M., 2006. *Link prediction using supervised learning*.In: SDM'06: Workshop on Link Analysis, Counter-terrorism and Security.
Almaatouq, A., Shmueli, E., Nouh, M., Alabdulkareem, A., Singh, V.K., Alsaleh, M., Alfaris, A., 2016. If it looks like a spammer and behaves like a spammer, it must be a spammer: analysis and detection of microblogging spam accounts. Int. J. Inf. Secur., 1–17.
Alsaleh, M., Alarifi, A., Al-Salman, A.M., Alfayez, M., Almuhaysin, A., 2014. *TSD: Detecting Sybil Accounts in Twitter*. In: 2014 Proceedings of the 13th IEEE International Conference onMachine Learning and Applications (ICMLA).
Apache, 2016. Apache Spark. Retrieved 20th January 2016, from⟨http://spark.apache.org/⟩
Aridhi, S., Lacomme, P., Ren, L., Vincent, B., 2015. A MapReduce-based approach for shortest path problem in large-scale networks. *Engineering Applications of Artificial Intelligence,41*, 151–165. doi: http://dx.doi.org/10.1016/j.engappai.2015.02.008
Arthur, D., Vassilvitskii, S., 2007. *k-means++: The advantages of careful seeding*. In: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms.
Balakrishnan, V., Humaidi, N., Lloyd-Yemoh, E., 2016. Improving document relevancy using integrated language modeling techniques. Malays. J. Comput. Sci. 29 (1).
Benevenuto, F., Rodrigues, T., Almeida, V., Almeida, J., Gonçalves, M.2009. Detecting spammers and content promoters in online video social networks. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval.
Bhat, S. Y., Abulaish, M.2013. Community-based features for identifying spammers in online social networks. In: Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining.
Bhat, S.Y., Abulaish, M., Mirza, A.A., 2014. Spammer classification using ensemble methods over structural social network features. 2014 Ieee/Wic/Acm Int. Jt. Conf. Web Intell. (Wi) Intell. Agent Technol. (Iat) 2, 454–458. http://dx.doi.org/10.1109/wi-iat.2014.133.
Bhattacharya, M., Islam, R., Abawajy, J., 2016. Evolutionary optimization: a big data perspective. J. Netw. Comput. Appl. 59, 416–426.
Bilge, L., Strufe, T., Balzarotti, D., Kirda, E.2009. All your contacts are belong to us: automated identity theft attacks on social networks. In: Proceedings of the 18th international conference on World wide web. ACM.
Boyd, D.M., Ellison, N.B., 2007. Social network sites: Definition, history, and scholarship. J. Comput. Commun. 13 (1), 210–230.
Cao, C., Caverlee, J.2015. Detecting Spam URLs in Social Media via Behavioral Analysis. Advances in Information Retrieval In: Proceedings of the 37th European Conference on IR Research, ECIR 2015, Vienna, Austria, March 29 - April 2, 2015. Proceedings, 703.
Cao, F., Ester, M., Qian, W., Zhou, A., 2006. Density-Based Clustering over an EvolvingData Stream with Noise. In:SDM.
Cao, J., Li, Q., Ji, Y., He, Y., Guo, D., 2016. Detection of Forwarding-Based Malicious URLs in Online Social Networks. Int. J. Parallel Program. 44 (1), 163–180.
Cao, Q., Sirivianos, M., Yang, X., Pregueiro, T.2012. Aiding the detection of fake accounts in large scale social online services. In: Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation.
Chan, P.P.K., Yang, C., Yeung, D.S., Ng, W.W.Y., 2014. Spam filtering for short messages in adversarial environment. Neurocomputing 155, 167–176. http://dx.doi.org/10.1016/j.neucom.2014.12.034.
Chen, C.-M., Guan, D., Su, Q.-K., 2014. Feature set identification for detecting suspicious URLs using Bayesian classification in social networks. Inf. Sci. 289, 133–147.
Chu, Z., Gianvecchio, S., Wang, H., Jajodia, S., 2012a. Detecting automation of twitter accounts: are you a human, bot, or cyborg? IEEE Trans. Dependable Secur. Comput. 9 (6), 811–824. http://dx.doi.org/10.1109/TDSC.2012.75.
Chu, Z., Wang, H., Widjaja, I., 2012b. *Detecting social spam campaigns on Twitter* (Vol. 7341 LNCS).
Clauset, A., Shalizi, C.R., Newman, M.E., 2009. Power-law distributions in empirical data. SIAM Rev. 51 (4), 661–703.
Conti, M., Poovendran, R., Secchiero, M.2012. FakeBook: Detecting Fake Profiles in On-Line Social Networks. In: 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM).
Cresci, S., Di Pietro, R., Petrocchi, M., Spognardi, A., Tesconi, M., 2015. Fame for sale: efficient detection of fake Twitter followers. Decis. Support Syst. 80, 56–71. http://dx.doi.org/10.1016/j.dss.2015.09.003.
Devineni, P., Koutra, D., Faloutsos, M., Faloutsos, C.2015, 25-28 Aug. 2015. If walls could talk: Patterns and anomalies in Facebook wallposts. In: 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM).
Diallo, O., Rodrigues, J.J., Sene, M., 2012. Real-time data management on wireless sensor networks: a survey. J. Netw. Comput. Appl. 35 (3), 1013–1021.
DMR, 2015. By The Numbers: 150+ Amazing Twitter Statistics. from⟨http://expandedramblings.com/index.php/march-2013-by-the-numbers-a-few-amazing-twitter-stats/10/⟩
Egele, M., Stringhini, G., Kruegel, C., Vigna, G., 2015. Towards Detecting compromised accounts on social networks. IEEE Trans. Dependable Secur. Comput.. http://dx.doi.org/10.1109/TDSC.2015.2479616.
Elyashar, A., Fire, M., Kagan, D., Elovici, Y.2013. Homing socialbots: intrusion on a specific organization's employee using Socialbots. In: Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining.
Ezpeleta, E., Zurutuza, U., Gomez Hidalgo, J. M.2015. An Analysis of the Effectiveness of Personalized Spam Using Online Social Network Public Information. In A. Herrero, B. Baruque, J. Sedano, H. Quintian E. Corchado (Eds.), International Joint Conference: Cisis'15 and Iceute'15 (Vol. 369, pp. 497–506.
Ferrara, E., Varol, O., Davis, C., Menczer, F., Flammini, A., 2014. The rise of social bots. *a*rXiv preprint arXiv:1407.5225.
Fire, M., Kagan, D., Elyashar, A., Elovici, Y., 2014. Friend or foe? Fake profile identification in online social networks. Soc. Netw. Anal. Min. 4 (1), 1–23.
Galán-García, P., de la Puerta, J. G., Gómez, C. L., Santos, I., Bringas, P. G.2014. Supervised Machine Learning for the Detection of Troll Profiles in Twitter Social Network: Application to a Real Case of Cyberbullying. In: International Joint Conference SOCO'13-CISIS'13-ICEUTE'13.
Gani, K., Hacid, H., Skraba, R.2012. Towards multiple identity detection in social networks. In: Proceedings of the 21st international conference companion on World Wide Web. ACM.
Gao, H., Hu, J., Wilson, C., Li, Z., Chen, Y., Zhao, B. Y.2010. Detecting and characterizing social spam campaigns. In: Proceedings of the 10th ACM SIGCOMM conference on Internet measurement.
Ge, J., Peng, J., Chen, Z.2014. Your Privacy Information are Leaking When You Surfing on the Social Networks: A Survey of the degree of online self-disclosure (DOSD). 2014 Ieee In: Proceedings of the 13th International Conference on Cognitive Informatics & Cognitive Computing (Icci-Cc), 329-336.
Ghosh, S., Viswanath, B., Kooti, F., Sharma, N.K., Korlam, G., Benevenuto, F., Gummadi, K.P., 2012. Understanding and combating link farming in the twitter social network. Proc. 21st Int. Conf. World Wide Web, 61.
Grier, C., Thomas, K., Paxson, V., Zhang, M.2010. @spam: the underground on 140 characters or less. In: Proceedings of the 17th ACM conference on Computer and communications security, 27-37.
Gupta, A., Kaushal, R., 2015. Improving Spam Detection in OnlineSocial Networks.
Harsule, S.R., Nighot, M.K., 2016. N-Gram Classifier System to Filter Spam Messages from OSN User Wall Innovations in Computer Science and Engineering. Springer, 21–28.
Hashem, I.A.T., Yaqoob, I., Anuar, N.B., Mokhtar, S., Gani, A., Khan, S.U., 2015. The rise of "big data" on cloud computing: review and open research issues. Inf. Syst. 47, 98–115 http://dx.doi.org/10.1016/j.is.2014.07.006.
Heidemann, J., Klier, M., Probst, F., 2012. Online social networks: a survey of a global phenomenon. Comput. Netw. 56 (18), 3866–3878 http://dx.doi.org/10.1016/j.comnet.2012.08.009.
Hu, X., Tang, J., Zhang, Y., Liu, H.2013. Social spammer detection in microblogging. In: Twenty-Proceedings of the Third international joint conference on Artificial Intelligence.
Igawa, R.A., Barbon, S., Jr, Paulo, K.C.S., Kido, G.S., Guido, R.C., Júnior, M.L.P., da Silva, I.N., 2016. Account classification in online social networks with LBCA and wavelets. Inf. Sci. 332, 72–83.
Javelin Strategy and Research, 2016. 2016 Identity fraud: Fraud hits an inlection point. Retrieved 3rd March 2016, from⟨https://www.javelinstrategy.com/coverage-area/2016-identity-fraud-fraud-hits-inflection-point⟩
Jiang, J., Shan, Z., Sha, W., Wang, X., Dai, Y.2012. Detecting and Validating Sybil Groups

in the Wild2012 In: Proceedings of the 32nd International Conference on Distributed Computing Systems Workshops (pp.127): IEEE. doi: 10.1109/ICDCSW.2012.9
Jin, L., Takabi, H., Joshi, J. B.2011. Towards active detection of identity clone attacks on online social networks. In: Proceedings of the first ACM conference on Data and application security and privacy.
Kaufman, L., Rousseeuw, P.J., 2009. Finding Groups in Data: An Introduction to Cluster Analysis 344. John Wiley & Sons.
Kharratzadeh, M., Renard, B., Coates, M.J., 2015. Bayesian topic model approaches to online and time-dependent clustering. Digit. Signal Process.http://dx.doi.org/10.1016/j.dsp.2015.03.010.
Kiruthiga, S., Kola Sujatha, P., Kannan, A., 2014. Detecting cloning attack in Social Networks using classification and clustering techniques. In: 2014 International Conference on Recent Trends in Information Technology (ICRTIT).
Kondratovich, E., Baskin, I.I., Varnek, A., 2013. Transductive support vector machines: promising approach to model small and unbalanced datasets. Mol. Inform. 32 (3), 261–266.
Kontaxis, G., Polakis, I., Ioannidis, S., Markatos, E.P., 2011. Detecting social network profile cloning. In: 2011 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops).
Lee, K., Caverlee, J., Webb, S.2010a. Uncovering Social Spammers: Social Honeypots plus Machine Learning. In: SIGIR 2010: Proceedings of the 33rd Annual International ACM SIGIR Conference on Research Development in Information Retrieval.
Lee, K., Caverlee, J., Webb, S.2010b. Uncovering social spammers: social honeypots+ machine learning. In: Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval.
Lee, S., Kim, J., 2014. Early filtering of ephemeral malicious accounts on Twitter. Comput. Commun. 54, 48–57.
Leskovec, J., 2015. Stanford large network dataset collection. from⟨https://snap.stanford.edu/data/⟩
Li, D.X., Fan, J.L., Wang, D.W., Liu, Y., 2012. Latent topic based multi-instance learning method for localized content-based image retrieval. Comput. Math. Appl. 64 (4), 500–510.
Li, Y., Xiao, R., Feng, J., Zhao, L., 2013. A semi-supervised learning approach for detection of phishing webpages. Opt.-Int. J. Light Electron Opt. 124 (23), 6027–6033 http://dx.doi.org/10.1016/j.ijleo.2013.04.078.
Lin, M.-S., Chiu, C.-Y., Lee, Y.-J., Pao, H.-K., 2013. Malicious URL filtering—A big data application. In: 2013 IEEE International Conference on Big Data.
Lin, P.-C., Huang, P.-M.2013. A study of effective features for detecting long-surviving Twitter spam accounts. 2013 In: Proceedings of the 15th International Conference on Advanced Communications Technology (ICACT), 841.
Liu, D., Mei, B., Chen, J., Lu, Z., Du, X., 2015. In: Dong, X.L., Yu, X., Sun, J. Li.Y. (Eds.), Community Based Spammer Detection in Social Networks 9098. Web-Age Information Management, 554–558.
Main, W., Shekokhar, N., 2015. Twitterati Identification System. In: Vasudevan, H., Joshi, A.R., Shekokar, N.M. (Eds.), *International Conference on Advanced Computing Technologies and Applications* (Vol. 45, pp. 32–41).
Markines, B., Cattuto, C., Menczer, F.2009. Social spam detection. In: Proceedings of the 5th International Workshop on Adversarial Information Retrieval on the Web.
Martinez-Romo, J., Araujo, L., 2013. Detecting malicious tweets in trending topics using a statistical analysis of language. Expert Syst. Appl. 40, 2992–3000. http://dx.doi.org/10.1016/j.eswa.2012.12.015.
Mccord, M., Chuah, M., 2011. Spam Detection on Twitter Using Traditional Classifiers Autonomic And Trusted Computing. Springer, 175–186.
Miller, Z., Dickinson, B., Deitrick, W., Hu, W., Wang, A.H., 2014. Twitter spammer detection using data stream clustering. Inf. Sci. 260, 64–73. http://dx.doi.org/10.1016/j.ins.2013.11.016.
Mislove, A., Viswanath, B., Gummadi, K. P., Druschel, P.2010. You are who you know: inferring user profiles in online social networks. In: Proceedings of the third ACM international conference on Web search and data mining.
Mohaisen, A., Yun, A., Kim, Y., 2010. *Measuring the mixing time of social graphs*. In: Proceedings of the 10th ACM SIGCOMM conference on Internet measurement. ACM.
Mulamba, D., Ray, I., Ray, I., 2016. *SybilRadar: A Graph-Structure Based Framework for Sybil Detection in On-line Social Networks*.In: IFIP International Information Security and Privacy Conference.
Narudin, F.A., Feizollah, A., Anuar, N.B., Gani, A., 2014. Evaluation of machine learning classifiers for mobile malware detection. Soft Comput., 1–15.
Nettleton, D.F., 2013. Survey: data mining of social networks represented as graphs. Comput. Sci. Rev. 7, 1–34. http://dx.doi.org/10.1016/j.cosrev.2012.12.001.
Newman, M.E., Girvan, M., 2004. Finding and evaluating community structure in networks. Phys. Rev. E 69 (2), 026113.
Nguyen, H., 2013. Research report 2013 state of social media spam.
Noriega, L., 2005. Multilayer perceptron tutorial. School of Computing, Staffordshire University.
Onnela, J.-P., Saramäki, J., Hyvönen, J., Szabó, G., Lazer, D., Kaski, K., Barabási, A.-L., 2007. Structure and tie strengths in mobile communication networks. Proc. National Acad. Sci. 104 (18), 7332–7336.
Ostrow, A., 2009. Koobface virus gets smarter; targets Twitter and Facebook user. from⟨http://mashable.com/2009/08/06/koobface-twitter-facebook/#zs_M9yTXKEqc⟩
Pappa, G.L., Ochoa, G., Hyde, M.R., Freitas, A.A., Woodward, J., Swan, J., 2014. Contrasting meta-learning and hyper-heuristic research: the role of evolutionary algorithms. Genet. Program. Evol. Mach. 15 (1), 3–35.
Pérez-Rosés, H., Sebé, F., Ribó, J.M., 2016. Endorsement deduction and ranking in social networks. Comput. Commun. 73, 200–210.
Quinlan, J.R., 2014. C4. 5: Programs for Machine Learning. Elsevier.
Ratkiewicz, J., Conover, M., Meiss, M., Gonçalves, B., Flammini, A., Menczer, F., 2011. *Detecting and Tracking Political Abuse in Social Media*.In: ICWSM.
Ruan, X., Wu, Z., Wang, H., Jajodia, S., 2016. Profiling Online Social Behaviors for Compromised Account Detection. *IEEE Trans. Inf.* FORENSICS *SECURITY* 11 (1), 176–187. http://dx.doi.org/10.1109/tifs.2015.2482465.
Sadan, Z., Schwartz, D.G., 2011. Social network analysis of web links to eliminate false positives in collaborative anti-spam systems. J. Netw. Comput. Appl. 34 (5), 1717–1723.
Sallaberry, A., Zaidi, F., Melançon, G., 2013. Model for generating artificial social networks having community structures with small-world and scale-free properties. Soc. Netw. Anal. Min. 3 (3), 597–609.
Savage, D., Zhang, X., Yu, X., Chou, P., Wang, Q., 2014. Anomaly detection in Online Social Networks. Soc. Netw. 39, 62–70. http://dx.doi.org/10.1016/j.socnet.2014.05.002.
Schaeffer, S.E., 2007. Graph clustering. Comput. Sci. Rev. 1 (1), 27–64.
Schneider, F., Feldmann, A., Krishnamurthy, B., Willinger, W., 2009. *Understanding online social network usage from a network perspective*. In: Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference.
Sethia, P., Karlapalem, K., 2011. A multi-agent simulation framework on small Hadoop cluster. Eng. Appl. Artif. Intell. 24 (7), 1120–1127 http://dx.doi.org/10.1016/j.engappai.2011.06.009.
Singh, M., Bansal, D., Sofat, S., 2014. Detecting Malicious Users in Twitter using Classifiers. ACM International Conference Proceeding Series, 247.
Singla, A., Patra, S., Bruzzone, L., 2014. A novel classification technique based on progressive transductive SVM learning. Pattern Recognit. Lett. 42, 101–106.
Smola, A.J., Schölkopf, B., 2004. A tutorial on support vector regression. Stat. Comput. 14 (3), 199–222.
Statista, 2016. Leading social networks worldwide as of April 2016, ranked by number of active users (in millions). from⟨http://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/⟩
Stein, T., Chen, E., Mangla, K., 2011. Facebook immune system In: Proceedings of the 4th Workshop on SocialNetwork Systems. ACM (pp.8): ACM. http://dx.doi.org/10.1145/1989656.1989664
Stringhini, G., Kruegel, C., Vigna, G., 2010. *Detecting spammers on social networks*. In: Proceedings of the 26th Annual Computer Security Applications Conference. ACM.
Tan, E., Guo, L., Chen, S., Zhang, X., Zhao, Y., 2013. *UNIK: unsupervised social network spam detection*. In: Proceedings of the 22nd ACM international conference on Conference on information & knowledge management.
Thomas, K., Grier, C., Song, D., Paxson, V., 2011. *Suspended accounts in retrospect: an analysis of twitter spam*.In: Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference.
Thomas, K., Nicol, D.M., 2010. *The Koobface botnet and the rise of social malware*. In: 2010 Proceedings of the 5th International Conference on Malicious and Unwanted Software (MALWARE).
Tran, N., Li, J., Subramanian, L., Chow, S.S., 2011. *Optimal sybil-resilient node admission control*.In:INFOCOM, 2011 Proceedings IEEE.
Twitter, 2016. The twitter rules. Retrieved 28th January 2016, from ⟨https://support.twitter.com/articles/18311⟩
Twitter rate limit, 2015. Twitter rate limit for search/tweets REST API calls. from⟨https://dev.twitter.com/rest/public/rate-limits⟩
Viswanath, B., Bashir, M.A., Crovella, M., Guha, S., Gummadi, K.P., Krishnamurthy, B., Mislove, A., 2014. *Towards detecting anomalous user behavior in online social networks*. In: Proceedings of the 23rd USENIX Security Symposium (USENIX Security)}.
Viswanath, B., Post, A., Gummadi, K.P., Mislove, A., 2011. An analysis of social network-based sybil defenses. ACM SIGCOMM Comput. Commun. Rev. 41 (4), 363–374.
Vlasselaer, V.V., Meskens, J., Van Dromme, D., Baesens, B., 2013. *Using social network knowledge for detecting spider constructions in social security fraud*.In: 2013 Ieee/Acm International Conference on Advances in Social Networks Analysis and Mining (Asonam).
Wang, A.H., 2010a. Detecting spam bots in online social networking sites: a machine learning approach *Data and Applications Security and Privacy XXIV* (pp. 335–342): Springer.
Wang, A.H., 2010b. *Do not follow me: Spam detection in twitter*. In:Security and Cryptography (SECRYPT), Proceedings of the 2010 International Conference on.
Wang, G., Mohanlal, M., Wilson, C., Wang, X., Metzger, M., Zheng, H., Zhao, B.Y., 2012a. Social turing tests: Crowdsourcing sybil detection. *arXiv preprint arXiv:1205.3856*.
Wang, G., Wilson, C., Zhao, X., Zhu, Y., Mohanlal, M., Zheng, H., Zhao, B.Y., 2012b. Serf and turf: crowdturfing for fun and profit. In: Proceedings of the 21st international conference on World Wide Web.
Wu, F., Shu, J., Huang, Y., Yuan, Z., 2016. Co-Detecting Social Spammers and Spam Messages in Microblogging via Exploiting Social Contexts. *Neurocomputing*.
Xin-fang, S., 2013. Survey of model and techniques for online social networks. In: 2013 Proceedings of the 8th International Conference on Computer Science & Education (ICCSE). IEEE.
Yang, C., Harkreader, R., Gu, G., 2013. Empirical Evaluation and New Design for Fighting Evolving Twitter Spammers. IEEE Trans. Inf. Forensics Security 8 (8), 1280–1293. http://dx.doi.org/10.1109/tifs.2013.2267732.
Yang, C., Harkreader, R.C., Gu, G., 2011. Die free or live hard? Empirical evaluation and new design for fighting evolving twitter spammers. In: Recent Advances in Intrusion Detection.
Yang, X., Wang, X., Tian, Y., Du, Y., 2015. Locality preserving based K-means clustering intelligence Science and Big data engineering. Big Data Mach. Learn. Tech., 86–95, (Springer).

Yang, Z., Wilson, C., Wang, X., Gao, T., Zhao, B.Y., Dai, Y., 2014a. Uncovering social network Sybils in the wild. Acm Trans. Knowl. Discov. Data 8 (1), 5–33. http://dx.doi.org/10.1145/2556609.
Yang, Z., Wilson, C., Wang, X., Gao, T., Zhao, B.Y., Dai, Y., 2014b. Uncovering social network Sybils in the wild. ACM Trans. Knowl. Discov. Data (TKDD) 8 (1), 2.
Yu, H., Gibbons, P.B., Kaminsky, M., Xiao, F., 2008. Sybillimit: A near-optimal social network defense against sybil attacks. In: IEEE Symposium on Security and Privacy, 2008. SP2008..
Yu, H., Kaminsky, M., Gibbons, P.B., Flaxman, A., 2006. Sybilguard: defending against sybil attacks via social networks. ACM SIGCOMM Comput. Commun. Rev. 36 (4), 267–278.
Zhang, R., Wang, W.J., Ma, Y.C., Men, C.Q., 2009. Least square transduction support vector machine. Neural Process. Lett. 29 (2), 133–142.
Zhang, Y., Lu, J., 2016. Discover millions of fake followers in Weibo. Soc. Netw. Anal. Min. 6 (1), 1–15.
Zheng, X., Zeng, Z., Chen, Z., Yu, Y., Rong, C., 2015. Detecting spammers on social networks. Neurocomputing 159, 27–34. http://dx.doi.org/10.1016/j.neucom.2015.02.047.
Yang, Z., Xue, J., Yang, X., Wang, X., Dai, Y., 2015. VoteTrust:leveraging friend invitation graph to defend against social network Sybils. IEEE Trans. Trans. Dependable Secur. Comput.. http://dx.doi.org/10.1109/TDSC.2015.2410792.
Zhu, X., Goldberg, A.B., 2009. Introduction to semi-supervised learning. Synth. Lect. Artif. Intell. Mach. Learn. 3 (1), 1–130.
Zuo, X., Blackburn, J., Kourtellis, N., Skvoretz, J., Iamnitchi, A., 2016. The power of indirect ties. Comput. Commun. 73, 188–199.

Adewole Kayode Sakariyah received B.Sc. and M.Sc degrees in Computer Science from University of Ilorin, Nigeria. He is an academic staff at the Department of Computer Science, Faculty of Communication and Information Sciences, University of Ilorin, Nigeria. Adewole is currently on his PhD program in the Department of Computer System & Technology, Faculty of Computer Science & Information Technology, University of Malaya, Malaysia. His Ph.D. research is in Network Security with specific focus on social networks. His research interests include Network Security, Biometrics, Machine learning, and Big Data analytics.

Nor Badrul Anuar obtained his Master of Computer Science from University of Malaya in 2003 and a Ph.D. at the Center for Information Security & Network Research, University of Plymouth, UK. He is a senior lecturer at the Faculty of Computer Science and Information Technology at University of Malaya, Kuala Lumpur. He has published a number of journal papers related to security areas locally and internationally. He has a good profile of publications in renowned Journals. His research interests include Intrusion Detection System (Intrusion Detection Systems, Intrusion Response Systems, Security Event and Management, Digital Forensic and Network Security), High Speed Network (Switching, Routing, IPV6, and Multicast) and Management Information System (E-thesis, Library Systems and Online Systems). He is also a member of IEEE Communications Society, IEEE Young Professionals and IEEE Computer Society.

Amirrudin Kamsin is a Senior Lecturer at the Faculty of Computer Science & Information Technology, University of Malaya, Malaysia. He received his BIT (Management) in 2001 and M.Sc. in Computer Animation in 2002 from University of Malaya and Bournemouth University, UK respectively. He obtained his Ph.D. from University College of London (UCL) in 2014. He has published a number of journal papers both locally and internationally. His research areas include human computer interaction (HCI), authentication systems, e-learning, mobile applications, serious game, augmented reality and mobile health services.

Kasturi Dewi Varathan is currently a senior lecturer at the Faculty of Computer Science & Information Technology, University of Malaya, Malaysia. She holds a Ph.D. in Computer Science from National University of Malaysia. She has published a number of journal papers both locally and internationally. Her research focus is in the field of Knowledge Representation and Information Retrieval mainly on sentiment analysis and opinion mining in social media.

Syed Abdul Razak is currently a senior lecturer at the Faculty of Arts and Social Sciences, University of Malaya, Malaysia. He holds a Ph.D. degree from University Adelaide Australia. He has published a number of journal papers both locally and internationally. His research focus is in the area of Social demography, population and social development.